

LEANDRO RODRIGUES FERREIRA

**ARPREM: AUTORIA DE REGRAS DE PRODUÇÃO PARA
REMEDIÇÃO DE ERROS COM MÚLTIPLAS
REPRESENTAÇÕES EXTERNAS**

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre. Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, Universidade Federal do Paraná.

Orientador: Prof. Dr. Andrey Ricardo Pimentel

CURITIBA

2015

LEANDRO RODRIGUES FERREIRA

**ARPREM: AUTORIA DE REGRAS DE PRODUÇÃO PARA
REMEDIÇÃO DE ERROS COM MÚLTIPLAS
REPRESENTAÇÕES EXTERNAS**

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre. Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, Universidade Federal do Paraná.

Orientador: Prof. Dr. Andrey Ricardo Pimentel

CURITIBA

2015

F383a

Ferreira, Leandro Rodrigues

ARPREM : autoria de regras de produção para remediação de erros com múltiplas representações externas/ Leandro Rodrigues Ferreira. – Curitiba, 2015.

145 f. : il. color. ; 30 cm.

Tese - Universidade Federal do Paraná, Setor de Ciências Exatas, Programa de Pós-graduação em Informática, 2015.

Orientador: Andrey Ricardo Pimentel .

Bibliografia: p. 96-101.

1. Sistemas tutoriais inteligentes. 2. Representação do conhecimento. 3. Estudantes - Avaliação. I. Universidade Federal do Paraná. II. Pimentel, Andrey Ricardo. III. Título.

CDD: 006.33

LEANDRO RODRIGUES FERREIRA

**ARPREM: AUTORIA DE REGRAS DE PRODUÇÃO PARA
REMEDIÇÃO DE ERROS COM MÚLTIPLAS
REPRESENTAÇÕES EXTERNAS**

Dissertação aprovada como requisito parcial à obtenção do grau de Mestre
no Programa de Pós-Graduação em Informática da Universidade Federal do
Paraná, pela Comissão formada pelos professores:

Orientador: Prof. Dr. Andrey Ricardo Pimentel
Departamento de Informática, UFPR

Profa. Dra. Maici Duarte Leite
Universidade Tecnológica Federal do Paraná campus
Francisco Beltrão, UTFPR

Prof. Dr. Alexandre Ibrahim Direne
Departamento de Informática, UFPR

Curitiba, 18 de novembro de 2015



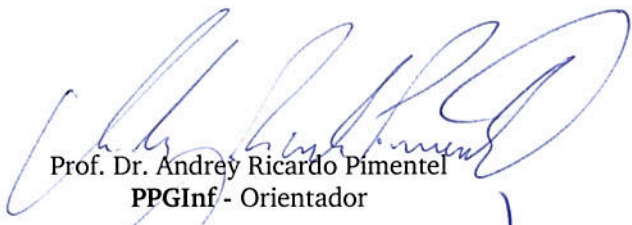
Ministério da Educação
Universidade Federal do Paraná
Programa de Pós-Graduação em Informática

PARECER


Nós, abaixo assinados, membros da Banca Examinadora da defesa de Dissertação de Mestrado em Informática, do aluno Leandro Rodrigues Ferreira, avaliamos o trabalho intitulado, “ARPREM: Autoria de Regras de Produção para Remediação de Erros com Múltiplas Representações Externas”, cuja defesa foi realizada no dia 18 de novembro de 2015, às 09:30 horas, no Departamento de Informática do Setor de Ciências Exatas da Universidade Federal do Paraná. Após a avaliação, decidimos pela:

☒ aprovação do candidato. () reprovação do candidato.

Curitiba, 18 de novembro de 2015.


Prof. Dr. André Ricardo Pimentel
PPGInf - Orientador


Prof. Dra. Maici Duarte Leite
UTFPR – Membro Externo


Prof. Dr. Alexandre Ibrahim Direne
PPGInf – Membro Interno



AGRADECIMENTOS

Ao Ser Supremo, à vida e a quaisquer outras formas de força superior que me permitiram a existência.

Ao Prof. Dr. Andrey Pimentel pela orientação e amizade, um ótimo orientador e por ter acreditado em meu trabalho e potencial.

Ao Prof. Dr. Alexandre Direne, por sempre ter sido uma referência e exemplo de docente, de pesquisador e de ser humano.

Aos meus pais Ana Cristina e Sergio, meus primeiros professores, pela educação, dedicação e sobretudo o amor incondicional, por terem feito o possível e o impossível para não nos deixar faltar nada. Amo vocês.

Ao meu companheiro David por ter me apoiado desde o início nos estudos e pela compreensão nos momentos mais cruciais da pesquisa. Amo você.

Aos meus irmãos Bruno, Matheus e Sergio Junior, que mesmo sendo os mais novos eu aprendo muito com vocês. Amo vocês maninhos.

Às minhas avós Benedita, Júlia e minha madrinha Lucy, que contribuíram para minha formação como cidadão, além de mostrar que laços fraternos estão acima de qualquer coisa.

À Profa. Dra. Laura García e Profa. Dra. Leticia Perez na qualificação pelas orientações e anotações que enriqueceram o trabalho final.

À Profa. Dra. Maici Duarte Leite que desenvolveu um ótimo trabalho, o qual me embasei para estendê-lo e senti lisonjeado por tê-la como avaliadora de meu trabalho.

À Rose Yuri, pela grande amizade desde os tempos de graduação e apoio por continuar os estudos.

À minha colega da linha de pesquisa e agora grande amiga Zenaide, pela grande amizade e apoio nos momentos cruciais.

Às minhas colegas da linha de pesquisa que tive contato Rita e Marcela, por toda a ajuda e exemplos de determinação.

Ao Diego Marczal por ter disponibilizado materiais de seus estudos que apoiaram e

inspiraram minha pesquisa.

À Alyne, minha primeira chefe e também grande amiga, que participou dos meus primeiros passos profissionais e mostrou que podemos sim fazermos grandes amigos no ambiente de trabalho.

Aos meus colegas e ex-colegas do CCE e DSI pela inspiração, pelos exemplos de grandes profissionais, pelas ajudas e que estavam à disposição quando sempre precisei: Antonio, Edson, Glaucio, Eduardo Manika, Sedimar, Rafael, Eduardo Nogueira, Elias, Sawamur, Alex, Leandro Nagano. Em especial a Denise, a Alexandra, a Pricila e o Giovanni, pelo incentivo, com dicas sobre o trabalho e pela grande amizade, pelas conversas confortantes nos momentos difíceis.

Aos demais colegas da UFPR, em especial Sonia Rocha, Sidney Max, Lilian Cristina, Claudia Monica Ritossa por ter gentilmente cedido a carta de recomendação, Samara, Fausto, Clovis, com os quais aprendi muitas coisas e sempre acreditaram no meu trabalho.

Aos demais colegas da graduação Rodrigo Cericatto, Leonardo Vidal, Andre Paulo Machado pelos bons momentos de estudos.

A todos meus professores do ensino básico até a graduação, que acrescentaram muito no meu processo de formação profissional e de cidadão. Sem vocês a sociedade simplesmente não poderia ser chamada de sociedade.

*É muito melhor lançar-se em busca de
conquistas grandiosas, mesmo
expondo-se ao fracasso, do que
alinhar-se com os pobres de espírito,
que nem gozam muito nem sofrem
muito, porque vivem numa penumbra
cinzenta, onde não conhecem nem
vitória, nem derrota.*

Theodore Roosevelt

SUMÁRIO

LISTA DE FIGURAS	viii
LISTA DE TABELAS	ix
RESUMO	x
ABSTRACT	xi
1 INTRODUÇÃO	1
1.1 Problema Central	2
1.2 Objetivos e Contribuição	2
1.3 Estrutura da Monografia	3
2 FUNDAMENTAÇÃO TEÓRICA	5
2.1 Sistemas Tutores Inteligentes	5
2.2 Representações Externas	9
2.3 Múltiplas Representações Externas	11
2.4 Regras de Produção	17
2.5 Adaptive Control of Thought - ACT	18
2.6 Conclusões	21
3 TRABALHOS CORRELATOS	22
3.1 Autoria de STIs	22
3.1.1 Apontamentos de Murray	22
3.1.2 Paradigmas	24
3.2 CTAT	26
3.3 SimStudent	27
3.4 ASSISTment	28
3.5 ASTUS	29

3.6	RUI	30
3.7	FARMA	31
3.8	Outros trabalhos de autoria de regras	31
3.9	Conclusões	32
4	ARQUITETURA DE REMEDIAÇÃO DE ERROS COM MÚLTIPLAS REPRESENTAÇÕES EXTERNAS PROPOSTA POR LEITE	34
4.0.1	Módulo Identificador de Expressões	34
4.0.2	Módulo Classificador de Erro	36
4.0.3	Módulo Classificador de Função de MRE	37
4.0.4	Gerenciador de MRE	37
4.1	Conclusões	39
5	ARPREM - AUTORIA DAS REGRAS DE PRODUÇÃO	41
5.1	Decisões de Implementação	41
5.2	Módulos da Arquitetura de Remediação de Erros	42
5.2.1	Interface do OA	43
5.2.2	Módulo Controlador Geral	43
5.2.3	Módulo Identificador de Expressões	44
5.2.4	Módulo Classificador do Erro	45
5.2.5	Módulo Classificador da Função da MRE	45
5.2.6	Módulo Gerenciador da MRE	46
5.2.6.1	Critérios de Remediação	46
5.3	Arquitetura para a Autoria das Regras de Produção	47
5.3.1	Módulo de Criação da Classificação de Erro	48
5.3.2	Módulo de Criação de MREs	49
5.3.3	Módulo de Criação de Exercícios	49
5.3.3.1	Definição da estrutura de resolução de um exercício	50
5.3.3.2	Criação de Caminhos de Resolução	52
5.3.3.3	Criação da Remediação	52

5.4	Esquema para Autoria das Regras de Produção	56
5.4.1	Autoria para as Respostas Corretas	58
5.4.2	Autoria para as Remediações dos Erros	59
5.4.2.1	Autoria das Regras para Respostas Erradas	61
5.4.2.2	Autoria das Regras para Classificador do Erro	62
5.4.2.3	Autoria das Regras para Classificador da Função MRE . .	62
5.4.3	Autoria para Gerenciador de MRE	66
5.4.4	Edição e Desativação das Regras de Produção	71
5.4.4.1	Edição das Regras	71
5.4.4.2	Desativação das Regras de Produção	72
5.5	Conclusões	72
6	ESTUDO DE CASO	74
6.1	Procedimento de Autoria	74
6.2	Análise das Saídas no Objeto de Aprendizagem	77
6.3	Estudo de Caso para o critério “Sucessos Anteriores com a MRE”	83
6.4	Estudo de caso para o critério “não mostrar MRE”	84
6.5	Estudo de caso para o critério “persistência no erro”	86
6.6	Estudo de caso para o critério “Alternância entre as MREs”	87
6.7	Conclusões	88
7	CONSIDERAÇÕES FINAIS	91
7.1	Trabalhos Futuros	93
	BIBLIOGRAFIA	96
	APÊNDICE	101
A	LISTA DAS REGRAS ACIONADAS NA EXECUÇÃO DO ESTUDO DE CASO	102

LISTA DE FIGURAS

2.1	Domínios que compõem um STI, adaptado de (39).	6
2.2	Arquitetura geral de um STI, adaptado de (39).	7
2.3	Taxonomia das Funções das Múltiplas Representações Externas, adaptado de (1)	16
4.1	Arquitetura Funcionalista de Remediação de Erros com MRE (25).	35
5.1	Exemplo de interface que representa um OA para estudo de caso da Arquitetura e autoria implementadas.	43
5.2	Fluxo de informações pela arquitetura de remediação de erros implementada.	48
5.3	Parte da tela para cadastro de classificador de erro, a ser utilizado posteriormente nas remediações.	49
5.4	Tela de cadastro de Múltipla Representação Externa.	50
5.5	Parte de um exemplo de caminho de resolução esperado pelo aprendiz. . .	51
5.6	Interface de autoria para o caminho de resolução de uma questão do exercício.	53
5.7	Tela de cadastro de remediação de erro.	55
5.8	Estrutura dos módulos apresentados e suas ligações.	56
5.9	Exemplos de estruturas das regras de produção para a autoria.	57
5.10	Esquema de autoria de uma regra de produção para resposta correta. . . .	59
5.11	Esquema de autoria de uma regra de produção para resposta errada. . . .	63
5.12	Esquema de autoria de uma regra de produção para Classificador do Erro.	64
5.13	Esquema de autoria de uma regra de produção para Classificador da Função MRE.	65
5.14	Parte da interface de autoria responsável pela interação do autor com a escolha do critério de remediação.	67
5.15	Estrutura das regras para os critérios de remediação “MRE Específica” e “Persistência no Erro”.	68

5.16	Estrutura das regras para os critérios de remediação “Não Exibir MRE Específica” e “Alternar Entre MREs”.	69
5.17	Estrutura das regras para os critérios de remediação “complexidade” e “sucessos anteriores”.	70
6.1	Aplicação no campo da Aritmética (24) e as identificações das MREs. A MRE 6 no caso será um texto introdutório à aritmética.	75
6.2	Parte da interface de autoria do exercício, com identificação dos campos e enunciado.	76
6.3	Interface de autoria do exercício e a exibição do caminho de resolução com os estados e metas gerados.	77

LISTA DE TABELAS

4.1	Composição da Classificação de erro e Função de MRE (24).	38
6.1	Relação das mediações cadastradas para o exemplo.	78
6.2	Relação de ações que simulam a resolução do exercício.	82
6.3	Relação de ações que simulam a resolução do exercício após edição de mediações.	83
6.4	Relação de ações que simulam a resolução do exercício após edição de mediações, para estudo do critério “não mostrar MRE específica”.	85
6.5	Relação de ações que simulam a resolução do exercício após edição de mediações, para estudo do critério “não mostrar MRE específica” e após inclusão de escolha aleatória.	85
6.6	Relação de ações que simulam a resolução do exercício após edição de mediações, para o critério “persistência no erro”.	86
6.7	Relação de ações que simulam a resolução do exercício após edição de mediações, para o critério “persistência no erro” e resolução aleatória de conflitos.	87
6.8	Relação de ações que simulam a resolução do exercício após edição de mediações para o critério “Alternância entre as MREs”.	88
6.9	Relação de ações que simulam a resolução do exercício após edição de mediações para o critério “Alternância entre as MREs” com aleatoriedade na resolução de conflitos entre as regras.	88

RESUMO

A utilização de Múltiplas Representações Externas em Sistemas Tutores Inteligentes, com a finalidade de elucidar conceitos do domínio a serem estudados pelo aluno, vem sendo explorada em vários estudos. Além disso, verificou-se que se aplicadas corretamente em quaisquer domínios estudados, as Múltiplas Representações Externas podem ser úteis no apoio ao aprendizado, em especial como itens importantes para remediar possíveis erros dos alunos durante o processo de resolução dos exercícios. Apoiados a esta motivação, alguns estudos abordam a relação entre as funções das Múltiplas Representações Externas e os tipos de erros cometidos pelo aluno, mediante uma classificação entre estes dois conceitos que possibilite a seleção das Múltiplas Representações Externas que melhor contribuam para a remediação do erro. É comum ferramentas de autoria para sistemas tutores inteligentes que sejam projetadas cuidadosamente, pois interferem em vários aspectos do resultado final, tais como a eficácia do sistema em passar o conhecimento, a autonomia de interferência do autor e do aluno, o nível de conhecimento do autor e o tempo de projeto do sistema. Múltiplas Representações Externas aliadas a uma arquitetura funcionalista de remediação de erros matemáticos juntamente com teorias cognitivas conceituadas, é proposta neste trabalho uma ferramenta de autoria das regras de produção integrada a esta arquitetura, de uma forma natural para professores sem conhecimento de programação. Após a contextualização da arquitetura, é apresentado um estudo de caso com uma simulação de autoria de um exercício e simulações de interações de estudantes, além de avaliar se as saídas esperadas das regras de produção são compatíveis com a autoria simulada.

ABSTRACT

The use of multiple external representations in intelligent tutoring systems, in order to elucidate domain concepts to be studied by the student, has been explored in several studies. In addition it was found that, if properly applied in any studied domains, the multiple external representations can be useful in supporting the learning, especially as important items to remedy possible errors of the students during the process of solving exercises. Relying on this motivation, some studies address the relationship between the functions of multiple external representations and the types of errors made by the learner through a rating between these two concepts that enables the selection of best multiple external representations that contribute to the error remediation. Authoring tools for intelligent tutoring systems should be designed carefully by interfering in various aspects of the final result, like system efficiency in transmitting knowledge, the author interference autonomy and learner, author knowledge level and design time of system. Multiple External Representations allied to a functionalist architecture mathematical errors remediation along with reputable cognitive theories, this work proposes an authoring tool of integrated production rules to this architecture, in a natural form to non-programmers. After the context of this architecture, a case study with a simulation of an exercise authoring and simulations of student interactions is showed, in order to evaluate whether the expected outputs of the production rules are compatible with the simulated authoring.

CAPÍTULO 1

INTRODUÇÃO

Atualmente tecnologias como ambiente web e máquinas portáteis, além dos tradicionais computadores *desktop*, possibilitam a utilização de aplicativos de diversas categorias. Entre estas categorias incluem os ambientes educacionais e jogos educativos, que estão cada vez mais disponíveis à população. Além disso, diversos objetos educacionais estão disponíveis gratuitamente, por iniciativa do Governo¹. No entanto, não há garantias que, na criação destes objetos, sejam utilizados módulos, padrões, conceitos comuns e itens importantes de um Sistema Tutor Inteligente (STI) aplicados a objetos de aprendizagem (OA). Vale mencionar, entre outros itens, técnicas de aquisição de conhecimento, agentes inteligentes, mecanismos para remediação de erros, possibilidade de exploração do conteúdo por parte do aprendiz ou a presença de um especialista do domínio para auxiliar no desenvolvimento do aplicativo. Sendo assim, a ausência destes recursos ou a desconsideração destes conceitos, assim como uma utilização incorreta, pode comprometer a qualidade e o objetivo do OA.

Partindo do mecanismo de remediação de erros, vários estudos apontam para a importância do aproveitamento do erro do aluno e para a vantagem de utilização dentro de um contexto de reparação e de aquisição do conhecimento (28; 24; 15). Em especial, Leite (24) propôs uma Arquitetura Funcionalista para Remediação de Erros, unindo em seu trabalho teorias para a aquisição de conceitos consolidadas como a Teoria de Classificações de Erros Matemáticos, Teoria das Múltiplas Representações Externas (MRE) (2) e Teoria ACT - “Adaptive Control of Thought” de John Anderson (6). Suas conclusões foram validadas em parceria com outra pesquisa, a “Ferramenta de Autoria para a Remediação de Erros com Mobilidade na Aprendizagem” (FARMA) (29; 31).

¹<http://objetoseducacionais2.mec.gov.br/>

Leite (24) também sugeriu como trabalho futuro a possibilidade de estruturar sua arquitetura em uma ferramenta de autoria com a finalidade de propor ao professor autonomia para atender à demanda de um grupo de alunos. Ferramentas de Autoria para STIs são aplicativos com suporte à construção de STIs, permitindo aos usuários sem conhecimento de programação construir seus próprios tutores e estruturar o conhecimento (37). Há diversas ferramentas de autoria em STIs desenvolvidas para atender a diferentes tipos de domínios e tarefas, como o CTAT (5), a FARMA (31), RUI (16) e o ASSISTment (45). Estas ferramentas podem trazer vantagens por garantir que o produto final (um OA) tenha todos os conceitos de um domínio, assuntos, tarefas e que cumpra o objetivo esperado. Outra vantagem destas ferramentas é garantir os recursos importantes de um STI, permitindo a autoria de seus módulos.

1.1 Problema Central

Dentre as ferramentas de autoria conhecidas, a “Cognitive Tutor Authoring Tools” (CTAT) (5), por exemplo, permite a criação e a edição de regras de produção. Porém, exige do autor conhecimento prévio da sintaxe da linguagem JESS ², no caso do CTAT, pois as regras são exibidas em sua sintaxe pura em um editor que destaca palavras reservadas e auxilia no controle dos fechamentos das regras. Também há editores de regras de produção que deixam o conteúdo dessas regras mais estruturado, por exemplo o jogo educacional *Renaissance* (50), porém ainda exigindo conhecimento por parte do professor em manipular estas regras. Além disso, estes trabalhos não contemplam autoria visando a remediação por MREs.

1.2 Objetivos e Contribuição

Este trabalho desenvolveu uma abordagem para a autoria das regras de produção da arquitetura de remediação de erros por Múltiplas Representações Externas, que possibilite ao professor adaptá-las às suas necessidades e que seja de fácil criação, em uma linguagem

²Em inglês, *Java Expert System Shell*.

natural, reduzindo a carga cognitiva dos autores no que se refere ao conhecimento de regras de produção, visto que nem todos os professores possuem domínio sobre regras de produção e como manipular estas regras adequadamente.

A abordagem de autoria das regras de produção pode contribuir com outros estudos para remediações de erros, levando em conta o impacto das Múltiplas Representações Externas através de seus formatos, complexidade e funções. Também, como contribuição, é esperada a expansão de estudos desta autoria para outras arquiteturas possíveis que venham a utilizar regras de produção. Além disso, pode-se pensar na integração e adaptação com ferramentas de autoria mais consolidadas, como a já mencionada FARMA (29; 31).

Os objetivos específicos necessários para alcançar o desenvolvimento desta abordagem de autoria são:

- implementar a arquitetura de remediação de erros matemáticos por MREs de Leite (24);
- planejar, modelar e implementar os módulos de autoria de regras de produção;
- efetuar testes em objeto de aprendizagem, que também será implementado como exemplo, e estruturar um estudo de caso;
- analisar as saídas e apresentar resultados destes testes.

1.3 Estrutura da Monografia

O documento está dividido nos seguintes capítulos: o Capítulo 2 apresenta um embasamento teórico sobre os conceitos utilizados para a proposta; o Capítulo 3 faz um levantamento de algumas pesquisas sobre autoria de sistemas tutores inteligentes; o Capítulo 4 explica a arquitetura de remediação de erros que é utilizada como base para esta pesquisa; o Capítulo 5 aborda a proposta deste trabalho, a ARPREM, que efetua a autoria das regras de produção, a integração com a arquitetura de remediação de erros

e como o autor irá interagir com estas regras de produção; o Capítulo 6 apresenta um estudo de caso e avaliação dos resultados da autoria proposta; e por último, o Capítulo 7 efetua as considerações finais e possíveis trabalhos futuros. Algumas informações geradas da execução do estudo de caso são incluídas nos anexos para maior apoio aos trabalhos futuros.

CAPÍTULO 2

FUNDAMENTAÇÃO TEÓRICA

Este capítulo tem como objetivo apresentar e elucidar os conceitos utilizados para o desenvolvimento deste trabalho.

2.1 Sistemas Tutores Inteligentes

Os Sistemas Tutores Inteligentes (STI)¹ são ferramentas educativas que visam auxiliar os professores e estudantes no processo de aquisição e transmissão do conhecimento (37). Mais que ferramentas educativas, incorporam técnicas de Inteligência Artificial (IA) e teorias de aprendizagem da psicologia. Como outra definição, STIs são programas de computador que são projetados para incorporar técnicas de IA, a fim de fornecer tutores que sabem *o que* eles ensinam, *para quem* ensinam e *como* ensinam (39).

A incorporação de Inteligência Artificial (AI) em técnicas de educação, com a finalidade de produzir educacionalmente artefatos computacionais remonta ao início dos anos 1970 (39). Desde a década de 1980 pesquisadores envolvidos nesta área começaram a se dividir em dois grupos de pensamento distintos. O primeiro e menor dos dois grupos defendeu os “ambientes de exploração”, que incentivaram a descoberta de aprendizagem (ou seja, aprender fazendo) e seus entusiastas se declararam como revolucionários por proporem mudanças drásticas nas metodologias tradicionais das escolas.

A linguagem LOGO foi o destaque desta linha de aprendizagem, que introduz os alunos para o mundo da Geometria através do uso de robôs “tartarugas” e de técnicas que o aluno aprende pela programação direta (39). Outra vertente de pesquisa, com mais adeptos, defendeu os próprios STIs, com uma visão mais reformista, ao trazerem uma proposta de melhoria gradual (evolução) na qualidade da educação ao incorporar IA. Este grupo

¹Do inglês *Intelligent Tutoring Systems* - ITS.

defendeu um paradigma onde o computador age como um tutor, ou seja, os alunos aprendem em grande parte ao receberem informações, o que é mais próximo às instruções de salas de aula tradicionais.

A concepção e o desenvolvimento desses tutores inteligentes estão situados na interseção da ciência da computação (IA), psicologia cognitiva e pesquisa educacional. Esta área de interseção é normalmente referida como a Ciência Cognitiva, conforme visto na figura 2.1.

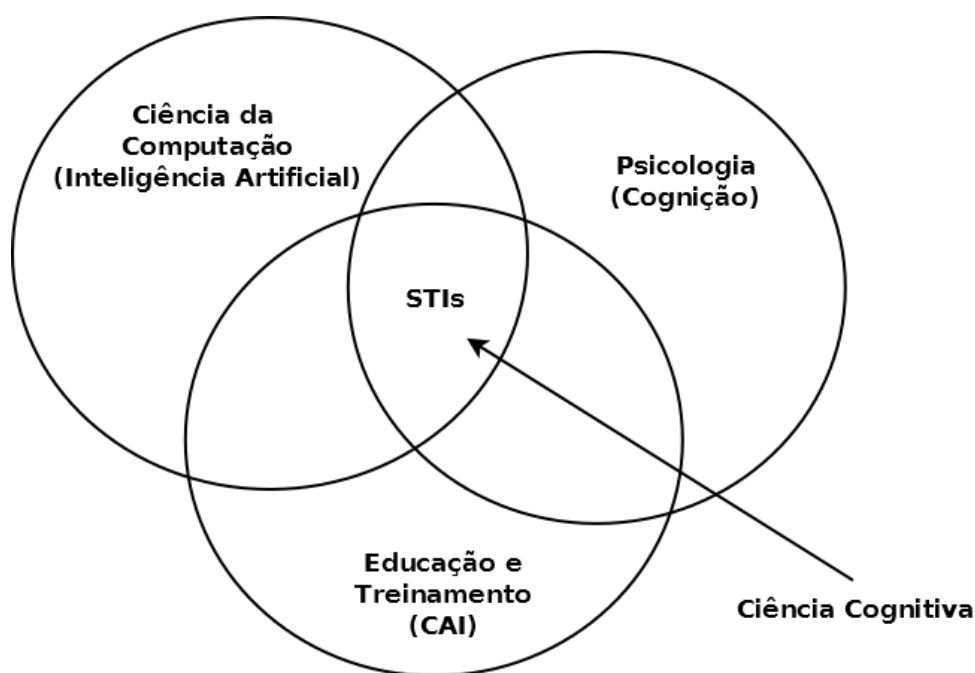


Figura 2.1: Domínios que compõem um STI, adaptado de (39).

Pode-se dizer que os STIs são uma evolução dos sistemas de Instruções Assistidas por Computador (CAI - *Computer Aided/Assisted Instruction*). Murray efetuou uma comparação entre CAI e STIs, mostrando que embora CAI sejam bem sucedidas para alguns tipos de aprendizagem e apresentar alguns conceitos como *feedback* e individualização, os STIs são mais robustos por conseguirem apresentar ao aprendiz instruções mais automatizadas próximas ao ideal natural de um tutor humano e mais adaptáveis para cada caso, menos estáticas pela incorporação de um modelo computacional do conteúdo (as bases de conhecimentos) (37; 38). Como desvantagens dos CAIs, ainda apresentavam, entre outros itens, a falta de articulação da representação de conhecimento, exibição estática

de conteúdos e uma arquitetura não muito bem definida (39).

Wenger (49) também aponta as vantagens dos STIS de flexibilidade no processo de ensino e estado de aprendizagem do aluno dos STIs, além de apresentar a arquitetura clássica de um STI (modelo do especialista do domínio, modelo pedagógico, modelo do estudante e interface), conforme visto na figura 2.2. Porém STIs podem variar muito no que se refere a sua arquitetura (39).

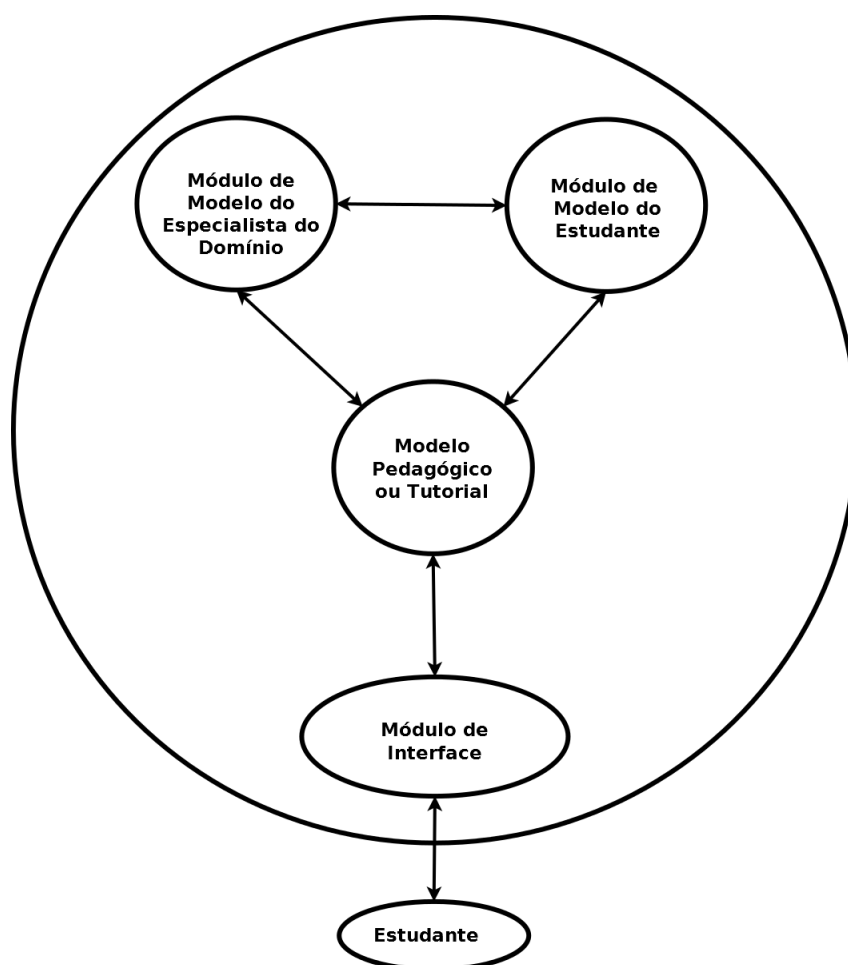


Figura 2.2: Arquitetura geral de um STI, adaptado de (39).

O Modelo de Domínio, módulo de conhecimento especializado do domínio, compreende os fatos e regras do domínio específico a ser transportado para o aluno, ou seja, o conhecimento dos especialistas (39). A principal diferença entre os STIs e sistemas *CAI* convencionais é que os STIs contêm este modelo inspecionável para o especialista do domínio (Modelo de Domínio) (37). Este conhecimento especializado é representado

de várias maneiras, incluindo redes semânticas, quadros e sistemas de produção. O Modelo de Domínio especifica *o que* será ensinado para os aprendizes (49). O modelo de domínio também contém dois tipos de informação: as informações de desempenho, que representam o conhecimento do estudante sobre o assunto e a resolução de problemas no domínio; e informação pedagógica (informação relevante para o aprendizado ou o ensino do conteúdo).

O Modelo do Aluno é o componente do sistema que mantém o controle de comportamentos de estudantes e infere o que o aluno sabe, representa o conhecimento e habilidade do estudante. Uma responsabilidade de um STI é ter essa compreensão do estudante. Em um caso ideal, esse modelo deve incluir todos os aspectos do comportamento e conhecimento do aluno que podem afetar seu desempenho e aprendizagem. No entanto, a tarefa de construir um modelo tão completo não é tão simples, especialmente considerando que o canal de comunicação, que é geralmente o teclado, pode ser restritivo. Tutores humanos normalmente combinam os dados a partir de uma variedade de outras fontes, como efeitos de voz ou gestos faciais. Eles também podem ser capazes de detectar outros fatores, tais como o tédio ou a motivação, que também são cruciais para a aprendizagem (39). Como exemplo de um modelo de aluno temos o “modelo de sobreposição do estudante” (49) em que o estado de conhecimento do estudante é visto como um subconjunto do domínio, com as respostas do estudante comparadas com os valores de domínio. Também pode-se citar “bibliotecas de erros” para mapear possíveis erros comuns do estudante.

O Modelo de Ensino² efetua o mapeamento das estratégias de ensino e como será o processo de aprendizagem do estudante, define o que será apresentado ao estudante conforme a resolução do exercício. Este modelo está fortemente ligado ao modelo do estudante, usando o conhecimento do aluno e sua estrutura tutorial para decidir quais as atividades pedagógicas serão apresentadas (dicas, apoio, explicações, etc.). O componente tutorial é, portanto, a fonte e o orquestrador de todas as intervenções pedagógicas.

²Também chamado de: Estratégia de Ensino, Modelo do Tutor ou Modelo Pedagógico.

O Módulo de Interface é responsável pela comunicação do estudante com o sistema, efetuando uma tradução entre a representação interna do sistema e uma linguagem de interface compreensível pelo estudante, ao processar o fluxo das informações de entrada e saída. Por isso projetistas de STIs não devem considerar a interface como um componente secundário, pois quando um STI apresenta um tópico, a interface pode melhorar ou atrapalhar a apresentação. Uma vez que a interface é a forma definitiva em que os STIs se apresentam para o estudante, qualidades como facilidade de uso e capacidade de atração podem ser cruciais para a aceitação do sistema por parte do estudante (39).

2.2 Representações Externas

Para o contexto deste trabalho, Representação Externa (RE) ³ é um termo para descrever o uso de técnicas para representar, apresentar e organizar o conhecimento (12). Consiste em uma ampla gama de representações nas modalidades linguística e gráfica, além da combinação destas modalidades. Zhang (51) definiu representações externas como o conhecimento e estrutura no ambiente, como símbolos físicos, objetos ou dimensões (por exemplo, símbolos escritos, contas de ábacos, dimensões de um gráfico, etc.), e como regras externas, restrições ou relações embutidas em configurações físicas (por exemplo, as relações espaciais de dígitos gravados, layouts visuais e espaciais de diagramas, restrições físicas em ábacos, etc.).

Em contraste, representações internas (RIs) são o conhecimento e estrutura na memória, como proposições, produções, esquemas, redes neurais, ou outras formas (51). As informações das RIs são recuperadas da memória através de processos cognitivos, mas as percepções de uma RE pode efetuar esta recuperação. Já as informações contidas nas REs podem ser coletadas, analisadas e processadas por percepções, embora a participação de conhecimento conceitual das RIs às vezes pode facilitar ou inibir os processos perceptivos. Para executar uma tarefa, as pessoas precisam processar a informação percebida das REs e as informações recuperadas a partir de RIs de forma entrelaçada, integradora

³Do inglês *External Representation* (ER).

e dinâmica. Por exemplo, considerando a tarefa de multiplicar os números 7 e 9. As RIs são os significados de cada um dos símbolos (por exemplo, o valor numérico do símbolo arbitrário “7” é sete), as tabelas de adição e multiplicação, procedimentos aritméticos, etc., que devem ser recuperados a partir da memória; as REs são as formas e posições dos símbolos, as relações espaciais de produtos parciais, etc., que podem ser inspecionadas perceptualmente do ambiente.

REs podem ser transformadas em RIs por memorização. Mas esta internalização não é necessária se as REs estão sempre disponíveis, e não é possível se as REs são muito complexas. RIs também podem ser transformadas em REs por externalização. Externalização pode ser benéfica se o benefício do uso de REs pode compensar o custo associado com o processo de externalização (51).

Como outros exemplos de RE temos sentenças em linguagem natural, sentenças em linguagens formais (por exemplo, lógica de primeira-ordem), tabelas, listas, grafos, mapas, projetos, diagramas, animações. Uma representação externa consiste em (2) apud (41):

- o mundo representado (informações fornecidas pela RE);
- o mundo representante (a representação do mundo, a forma como é apresentada a RE);
- quais os aspectos do mundo representado que estão sendo expressos;
- quais os aspectos do mundo representante que compõem a modelagem; e
- a correspondência entre os dois mundos.

Tanto o mundo representado quanto o mundo representante devem ser considerados ao levar em conta a eficácia da representação. As REs estão presentes no cotidiano, por exemplo em anotações, lembretes, placas de sinalização, resolução de contas com papel e caneta, lista de compras, entre outros. Para Zhang (51), apesar das REs auxiliarem o processo de memorização com a diminuição da carga cognitiva para resolver um problema,

tornando esta característica a mais evidente, algumas tarefas simplesmente não existiriam sem as REs, ou mudariam radicalmente sua forma de resolução. As REs também permitem o acesso de conhecimentos e habilidades que não estão disponíveis a partir de RIs. Por isso não se deve considerar que as REs contribuem apenas como auxílio à memória. Diferentes REs de uma mesma informação podem causar diferentes comportamentos cognitivos e alterar estratégias de tomada de decisão. Entre as tarefas cognitivas com apoio das REs, podemos mencionar resolução de problemas, criação de novos conhecimentos, raciocínio ou tomada de decisão.

2.3 Múltiplas Representações Externas

Vários estudos apontam as vantagens em utilizar REs apropriadas no apoio à aprendizagem, melhorando o desempenho dos estudantes (51; 1; 2; 3). Ainsworth apresenta um estudo que defende o uso de mais de uma representação ou de representações dinâmicas, propondo a aprendizagem com Múltiplas Representações Externas (MREs)⁴ (3). MREs podem proporcionar benefícios na aprendizagem de conhecimentos novos e complexos. Porém, alerta para o cuidado na forma de apresentação das MREs, pois apesar de as MREs serem ferramentas poderosas com potencial para contribuir para o aprendizado, sua combinação ou apresentação de forma errônea pode levar os alunos a não conseguirem explorar suas vantagens, sendo que nos piores casos a aprendizagem pode ser prejudicada. Assim, é necessário considerar as circunstâncias que influenciam na eficácia das MREs, principalmente em domínios complexos que envolvam diversas variáveis e processos (19).

Ainsworth apresenta uma abordagem com diferentes aspectos da aprendizagem com representações, sugerindo uma série de fatores na construção das MREs. O DeFT (Projeto, Funções, Tarefas)⁵ (2) é um arcabouço para a aprendizagem através das múltiplas representações, que integra os estudos sobre aprendizagem, da área da ciência cognitiva e das teorias construtivistas da educação. A contribuição desta estrutura consiste em propor a identificação de uma ampla gama de fatores que influenciam a aprendizagem,

⁴Do inglês *Multiple External Representations (MERs)*.

⁵Do inglês *Design, Functions, Tasks*.

conciliando descobertas experimentais que anteriormente se encontravam inconsistentes, determinando aspectos não muito explorados através da aprendizagem com múltiplas representações e avançando na criação de um projeto heurístico para este tipo de aprendizagem. O DeFT sugere que muitas dimensões se combinem para indicar se um aluno será capaz de se beneficiar da aprendizagem com uma combinação específica de representações. As dimensões consideradas no DeFT são:

- os parâmetros de projeto para a aprendizagem utilizando as múltiplas representações, são únicos para a aprendizagem com mais de uma representação (número de representações, a forma que a representação é distribuída, forma do sistema representacional, sequência de representações e suporte para tradução entre as representações) (*Design*);
- as funções pedagógicas que servem para apoiar a aprendizagem (*Functions*);
- e as tarefas cognitivas que devem ser realizadas e compreendidas pelo estudante ao interagir com a estratégia da aprendizagem fornecida pela MRE (por exemplo, o estudante deve entender a forma da representação, a relação entre a representação e o domínio e como selecionar uma representação apropriada, se for o caso) (*Tasks*).

Há um conjunto de dimensões do projeto que se aplicam exclusivamente aos sistemas multi-representacionais:

- *número*: o número de representações empregadas. Por definição, os sistemas multi-representacionais empregam pelo menos duas representações. Geralmente muitas representações estão disponíveis num sistema ou de forma simultânea ou em algum ponto durante a interação dos alunos com elas. No entanto, um número excessivo de representações raramente contribui para a aprendizagem;
- *informação*: a forma como a informação é distribuída ao longo das representações. Sistemas multi-representacionais podem permitir uma flexibilidade na maneira como a informação é distribuída entre as representações. Logo, esta flexibilidade colide

com a complexidade de informações em cada uma e a redundância de informação entre todas elas. Num dos extremos, pode transmitir a representação de cada conteúdo de forma completamente diferente (como se vê na representação de dois mundos) e, neste caso, não há redundância entre representações. A distribuição de informações desta forma pode simplificar cada representação individual, mas exige representações adicionais. Os alunos necessitam integrar informações de várias fontes. Os sistemas podem também ser parcialmente redundantes, de modo que algumas das informações são constantes entre as representações. Por fim, cada representação poderia ser concebida para expressar a mesma informação e assim a única diferença entre as representações é nas suas propriedades computacionais (mundos em representação). Neste caso, há redundância de informação completa em todo o sistema, mas muitas vezes uma representação é mais complexa;

- *forma*: a forma do sistema de representação. Um sistema pode apresentar imagens, textos, animações, som, equações e gráficos, simultaneamente ou não, mas uma questão é saber se a apresentação é viável. Por conseguinte, muito ainda é desconhecido sobre como um sistema representacional influencia a aprendizagem. Para entender a aprendizagem com MRE não é suficiente considerar cada tipo de representação isoladamente, pois representações interagem umas com as outras. Como resultado, há um espaço potencialmente vasto para explorar;
- *sequência*: a sequência de apresentação das representações. Se nem todas as representações são desenhadas em simultâneo, uma série de outras questões surgem. A primeira questão é a sequência em que as representações devem ser apresentadas ou construídas. Mesmo se a sequência foi pre-determinada, o aluno ou o sistema ainda precisa decidir em que ponto adicionar a nova representação ou alternar entre as representações;
- *tradução*: apoio para a tradução entre representações. Ambientes informatizados permitem uma ampla variedade de maneiras de indicar aos alunos a relação entre as representações. Duas dimensões têm recebido alguma atenção. Em primeiro lugar,

o papel ativo que o ambiente desempenha no apoio a alunos. Em segundo lugar, o apoio fornecido a nível sintático ou a nível semântico.

Ainsworth também destaca que as representações podem contribuir para a aprendizagem a partir de suas funções, após efetuar uma análise conceitual de ambientes de aprendizagem multi-representacionais. Foi então proposta uma taxonomia para funções das MREs, com três grandes funções principais: complementar, restringir e construir. Esta taxonomia de funções pode ser visualizada na figura 2.3 (1).

A função *Papéis Complementares* tem a finalidade de explorar a representação para apoiar ou complementar um processo cognitivo. MREs se complementam porque se diferem tanto nos processos que cada uma suporta ou nas informações que cada uma contém. Através da combinação destas representações, espera-se que os alunos se beneficiem da soma das vantagens de cada uma. Esta função pode ser dividida em duas subfunções:

- **Informações Complementares:** acontece quando as MREs utilizadas contém informações diferentes. Geralmente ocorre quando uma representação sozinha pode ser muito complicada, ou se apresentar toda a informação de um conceito relativamente complexo, ou também se informações apresentam-se em escalas diferentes. Por exemplo, REs gráficas nem sempre são melhores, pois alguns alunos consolidam sua cognição melhor com gráficos, outros com textos. Dependendo da tarefa, essa subfunção possui duas subdivisões: *Informações Diferentes*, que agregam informações distintas; e *Informações Compartilhadas*, que possuem algumas informações em comum.
- **Processos Complementares:** ocorre quando as REs possuem a mesma informação, mas diferem nas vantagens da aprendizagem, ou seja, nas necessidades específicas de cada situação. Possui três subdivisões: *Tarefa*, quando as MREs coincidem com as estruturas das informações da tarefa, facilitando o desempenho; *Diferenças Pessoais*, que possibilita o aluno escolher a que melhor atenda às suas necessidades; e *Estratégia*, onde diferentes formas de REs ajudam a traçar estratégias mais ou

menos eficazes, com o intuito de utilizar mais de uma estratégia para resolver o mesmo problema, onde várias REs são melhores que uma sozinha (se uma RE possui alguma falha em sua estratégia, a outra complementa o entendimento).

A função de *Restrição* é útil quando uma RE restringe a interpretação da outra. É dividida em duas subfunções:

- Familiaridade: A familiaridade dos alunos com uma RE restringe a necessidade de interpretação de outra menos conhecida. Por exemplo, simulações concretas ao lado de gráficos mais complexos.
- Herança: quando a interpretação de uma RE genérica pode ser restringida por uma segunda RE mais específica.

Nas funções de *Compreensão Mais Aprofundada* as REs apoiam a construção de compreensão quando usadas em conjunto, e a percepção alcançada pode ser transferida para outras situações. Muitas funções dependem do conhecimento do aluno. Por exemplo, o aluno familiarizado com tabelas pode ampliar conhecimento com gráficos (extensão). Outro aluno possui familiaridade com ambas as representações, mas desconhece a relação entre elas (relação). Esta função apresenta três subdivisões:

- Abstração: alunos criam entidades mentais para a base de conceitos e processos em um nível superior de organização .
- Extensão: ampliar o conhecimento a partir de um conhecimento adquirido anteriormente para algo desconhecido. Um gráfico velocidade-tempo pode ampliar conhecimentos para aceleração.
- Compreensão Relacional: compreender a relação entre as duas representações.

Vale frisar algumas diferenças observadas entre as funções, com a finalidade de facilitar a compreensão desses conceitos:

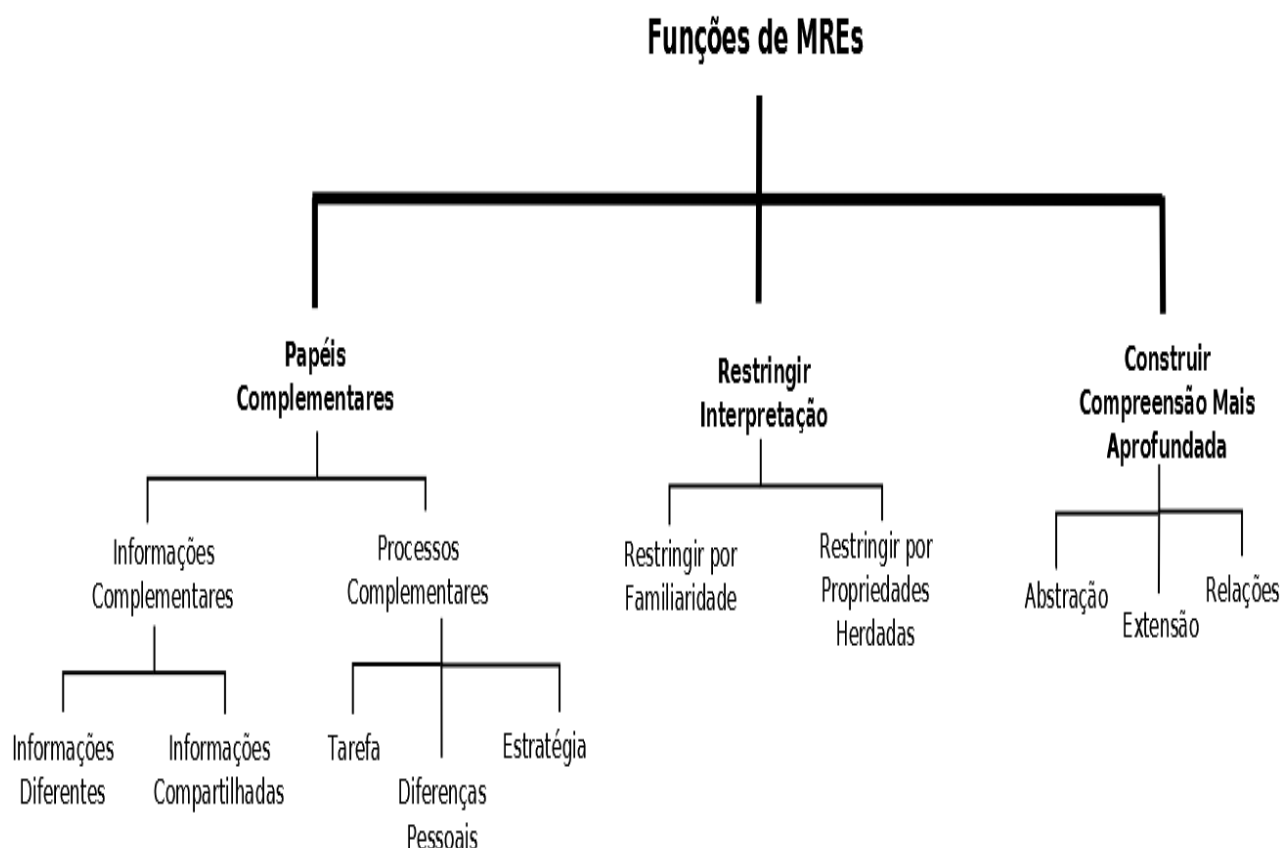


Figura 2.3: Taxonomia das Funções das Múltiplas Representações Externas, adaptado de (1)

- Diferença entre Padrões Complementares e Restrição de Interpretação: o Padrão Complementar é necessário sempre e independente se o aluno é mais “esperto” ou não, e Restrição de Interpretação pode ser retirada se o aluno entendeu o real significado da primeira representação.
- Diferença entre Construção de Conhecimento e Restrição de Interpretação: na Construção do Conhecimento todas as REs contribuem para o entendimento do domínio, enquanto na Restrição de Interpretação uma representação apoia a compreensão da outra.

Ainsworth, além de mostrar que as MREs podem oferecer essas três funções para apoiar a aprendizagem, mostrou também que a função que uma MRE pode suportar não depende apenas da intenção do projetista, mas também do conhecimento e objetivo do aluno. Um outro fator que deve ser considerado é que qualquer combinação particular de represen-

tações podem também suportar múltiplas funções simultaneamente (2; 3). Por exemplo, um ambiente pode representar a velocidade através do uso de uma tabela, uma equação, uma exposição numérica, uma animação ou um gráfico. Isso permite vantagens proporcionadas pelas diferentes propriedades complementares a estas formas de representação, com informações diferentes que elas fornecem (por exemplo, alterações ao longo do tempo). Podem também proporcionar facilidade de interpretação e familiaridade com números e animação, para ajudar os alunos a compreender uma representação desconhecida, restringindo como eles podem interpretá-las. Finalmente, podem ajudar os alunos a construir uma compreensão mais aprofundada da força e movimento, ajudando-os a formar abstrações sobre vários casos, ou relacionar e aumentar seu conhecimento a partir de tabelas e gráficos, por exemplo.

2.4 Regras de Produção

As regras de produção são pequenas unidades de conhecimento relativamente independentes, mesmo trabalhando em conjunto com outras regras, permitindo assim que o conhecimento possa ser construído de forma incremental (48). As regras de produção têm sua origem em pesquisas lógico-matemáticas sobre a possibilidade de escrever procedimentos através de uma sequência de regras (24) apud (35). São utilizadas para representar um conhecimento heurístico sobre o mundo e especifica um conjunto de ações que devem ser realizadas para uma dada situação.

Uma regra é composta por uma parte de condições (a parte “se” da regra, ou lado esquerdo da regra, ou LHS – de *left-hand size*) e uma parte de ações (a parte “então” da regra, ou lado direito da regra, ou RHS – de *right-hand size*). Uma condição apresenta uma lista de símbolos ou variáveis, que devem ser unificadas com os fatos da memória de trabalho. O processo de unificação ⁶ é feito por um motor de inferência.

⁶Do inglês *pattern matching*.

Um sistema de produção é uma denominação genérica de um sistema que utilizam as regras de produção para representação do conhecimento. O sistema de produção é composto por um conjunto de regras de produção, juntamente com uma memória de trabalho/declarativa (onde são localizados os fatos) e um algoritmo chamado encadeamento progressivo ⁷ que gera novos fatos a partir de fatos já existentes. A regra é disparada se suas condições forem atendidas pelos fatos da memória de trabalho. Nos sistemas dedutivos, todas as regras que estão habilitadas podem disparar, gerando novas asserções na memória de trabalho. Já os sistemas reativos utilizam uma estratégia de resolução de conflitos, que examina qual regra será disparada dentro de um conjunto de conflitos (ou seja, conjunto das regras prontas para serem disparadas). Conforme as ações executadas, este conjunto de conflitos pode ser modificado, com o acréscimo ou remoção de fatos, em um ciclo que se repete até que não haja mais regras disparáveis.

Na teoria ACT, descrita a seguir, o conhecimento procedimental representa suas informações em forma de regras de produção. Conforme dito anteriormente, as regras de produção são compostas por fatos, condições e ações baseados na memória declarativa. As condições de uma regra apresentam símbolos ou variáveis que devem ser unificadas com os fatos, através de um motor de inferência.

2.5 Adaptive Control of Thought - ACT

ACT (*Adaptive Control of Thought*) é uma teoria geral de cognição proposta e desenvolvida por John Anderson na *Carnegie-Mellon University*, que possui como objeto de estudo os processos de memória. Além disso, o ACT é uma arquitetura cognitiva, ou seja, uma teoria para simular e entender a cognição humana. ACT-R é a atual versão de uma série de modelos cognitivos da família ACT, sucedendo ACT, ACTE e ACT*. Com ACT* (6) Anderson se preocupou em descrever as formas como os mecanismos de recuperação de memória e cognição ocorrem, as formas como as metas são divididas em submetas através das regras de produção. Depois, Anderson questionou a maneira como

⁷Do inglês *forward-chaining*.

processos cognitivos, tais como memória, categorização, inferência casual e resolução de problemas poderiam ser selecionados como boas soluções para tarefas humanas comuns. Partindo da crença de que numa perspectiva evolucionária os seres humanos procuram se adaptar ao máximo ao meio ambiente, isto também acontece na recuperação de memória e na tomada de decisões, que tendem a ser otimizadas. Esta otimização restringe os possíveis mecanismos de cognição humana e, com isso, ajuda a reduzir o espaço de busca para o mecanismo real. Com isso, Anderson revisou o ACT* para criar o ACT-R, onde o “R” (racional) implica que a mente humana estará se tornando racional, para resolver problemas maximizando benefícios e minimizando custos.

ACT-R (*ACT Rational*) modela a maneira como os humanos recuperam itens de informação da memória e como eles resolvem problemas ao dividi-los em submetas e aplicar conhecimento da memória de trabalho quando necessário. A finalidade de um modelo ACT-R é interpretar comportamentos do estudante (47) e, para isso, o modelo precisa interpretar corretamente a representação do conhecimento humano e também entender como aquele conhecimento resulta em comportamentos específicos. Em outras palavras, a teoria do ACT-R procura entender como as pessoas organizam o conhecimento e como produzem comportamentos inteligentes. No campo educacional, esta representação do conhecimento resulta em previsões sobre o que os alunos podem e não podem fazer, bem como quais atividades e experiências irão auxiliar os alunos a atingirem metas de aprendizagem.

Essa teoria se fundamenta em manter o aluno em uma linha de aprendizagem ideal baseado em 3 princípios (6).

- O primeiro princípio do ACT-R sugere a divisão do conhecimento humano em dois tipos básicos de conhecimento: o Conhecimento Declarativo e o Conhecimento Procedural. O *Conhecimento Declarativo* é o conhecimento que podemos descrever, que temos consciência de como fazer. Podemos relacionar este tipo de conhecimento ao conjunto de fatos, aos muitos pedaços do conhecimento e às metas que estão ativas. Como exemplos de fatos temos “A escola está fechada aos sábados” e “A meta atual

é ir para casa”, que podem referenciar a outros fatos, como conceitos de “escola”, “fechada”, “sábado” que se relacionam entre si. O *Conhecimento Procedural* (ou *Procedimental*) é o conhecimento ligado ao nosso comportamento sem a necessidade de nossa plena consciência. Ocorre quando temos compreensão da tarefa de uma forma natural. Este conhecimento está ligado ao conjunto de regras de produção que especificam como uma determinada meta pode ser alcançada quando uma pré-condição é atendida. Uma regra de produção pode indicar, por exemplo, “Se é sábado, então não vou para a escola”.

- O segundo princípio sugere que tarefas complexas necessitam da combinação dos dois tipos básicos de conhecimento, tanto do procedimental quanto do declarativo.
- O terceiro princípio sugere que os conhecimentos mais utilizados serão lembrados ou recuperados mais facilmente. Os conhecimentos menos utilizados tendem a uma recuperação mais difícil e, em alguns casos, esta recuperação chega a ser impossível.

Além disso, o ACT apoia o fato de que as habilidades cognitivas possam ser modeladas independentemente de regras de produção que indicam o objetivo da resolução do problema, tendo dois modelos que atendem a este fato: o modelo de rastreamento (do aluno) e o modelo do conhecimento rastreado. O modelo de rastreamento (*model tracing*) acompanha o aluno individualmente no caminho da resolução de um problema, oferecendo suporte ao não concordar com a resposta do aluno em uma determinada etapa de resolução do problema. Segue as ações do estudante para determinar a estratégia particular do estudante em resolver o problema. O modelo do conhecimento rastreado (*knowledge tracing*) implica traçar o caminho da solução para cada resposta do aluno na etapa corrente, fazendo com que o tutor estime o conhecimento do aluno conforme a regra acionada através de sua resposta. Cada ação do estudante é associada a uma ou mais habilidades, as quais fazem referência aos componentes do conhecimento no modelo cognitivo.

2.6 Conclusões

O presente capítulo apresentou a base teórica dos conceitos que serão utilizados para a proposta de autoria das regras de produção. A arquitetura de remediação de erros proposta por Leite (24) utiliza teorias para aquisição de conceitos consolidadas que suportam a aprendizagem, por exemplo a Teoria das MREs (2) e Teoria ACT (6), além de utilizar Regras de Produção como parte importante da arquitetura que irá efetuar as classificações necessárias. A interface para a autoria destas regras de produção utilizará os princípios do ACT, principalmente o Conhecimento Procedural para a construção da remediação do erro, além de um apoio para demonstrar o modelo de rastreamento do aluno.

Embora alguns itens não pertençam ao foco deste trabalho que é a autoria das regras de produção, como forma de composição das MREs e sua classificação quanto aos atributos de projeto (*Design*), foi necessário inseri-los para suportar um melhor entendimento do trabalho.

CAPÍTULO 3

TRABALHOS CORRELATOS

Este capítulo apresenta uma revisão bibliográfica de trabalhos referentes a ambientes de autoria de STIs e de Regras de Produção.

3.1 Autoria de STIs

3.1.1 Apontamentos de Murray

Murray (36) aborda a necessidade de ferramentas e técnicas de autoria para STIs, descrevendo ferramentas para a criação de aspectos de um tutor com base em conhecimento.

Segundo Murray (37) tutores humanos têm mais de um método de ensino ou estilo à sua disposição, e do mesmo modo, os tutores de computador devem ser capazes de mudar o estilo de ensino, dependendo de características dinâmicas dos alunos. Murray ainda recomenda que Técnicas de Inteligência Artificial, tais como reconhecimento de plano, raciocínio baseado em casos, compreensão da linguagem natural, redes neurais, lógica fuzzy e baseadas em regras de resolução de problemas devem ser usadas com moderação e cautela, se o objetivo é produzir tutores de forma eficiente, uma vez que a inclusão dessas tecnologias em sistemas de tutoria aumenta drasticamente a complexidade das ferramentas de criação necessárias, a quantidade de treinamento necessário para usá-los, bem como a complexidade de aquisição de conhecimento.

Murray (37; 38) também efetuou uma análise de vários ambientes de autoria de STIs ao procurar classificá-los de acordo com alguns critérios como quais tipos de tutores, domínios ou tarefas podem ser construídos e quais métodos e ferramentas são utilizados para facilitar a autoria. Além disso, foi apresentado se os ambientes de autoria foram usados em situações reais e se foram passados por avaliações.

Foram elencados ferramentas e métodos utilizados por sistemas de autoria para simplificar e automatizar a própria autoria e aquisição do conhecimento, cuja meta principal (entre outras metas, a mais importante ou a mais predominante) é a diminuição dos esforços (em tempo, custo e/ou outros recursos) na criação de tutores inteligentes. As demais metas são as seguintes:

- diminuição de habilidades necessárias para a construção de STIs (facilita o acesso de mais pessoas a fazerem parte do processo de criação);
- auxiliar o autor a organizar o conhecimento do domínio ou pedagógico;
- apoiar bons princípios de projeto (modelos pedagógico, interface com o usuário, etc.);
- permitir prototipagem rápida dos projetos desses tutores (ciclos de avaliação/projeto dos protótipos).

Para alcançar estas metas, Murray apontou os seguintes métodos que facilitam a autoria de STIs:

- Articulação do conhecimento estruturado com modelos: diminuição de tarefas ao incorporar um modelo ou um arcabouço particular; restrição do universo que pode ser construído; apoiar o autor com modelos e arcabouços que ajudem a organizar e estruturar a informação.
- Conhecimento incorporado e conhecimento padrão: incorporação de conhecimento prévio na ferramenta de autoria; pode proporcionar autoria mais fácil com instruções de apoio à construção.
- Gerenciamento do conhecimento: procurar facilitar a manipulação das informações entre os módulos de um STI e o relacionamento entre eles; técnicas de interface com o usuário e de navegação permitem auxiliar os autores na manipulação destas informações.

- Visualização do conhecimento: ajudar os autores a entenderem e compreenderem toda a complexidade do conhecimento através de boas ferramentas de visualização, como simulação de cenários, diagramas, entre outros.
- Elicitação do conhecimento e gerenciamento de linha de trabalho: preocupação em obter o conhecimento de domínio do especialista; por exemplo diálogos interativos com o autor podem ajudar neste processo de aquisição do conhecimento.
- Validação do conhecimento e projeto: verificação das informações passadas pelo autor, se estão consistentes, completas, precisas; sistemas de autoria tendem a deixar as entradas passadas pelo autor mais flexíveis, da forma que ele achar mais viável e com isso algumas informações podem ser inconsistentes, imprecisas ou ir contra boas práticas de projeto.
- Reutilização do conhecimento: permitir a reutilização de elementos de um STI para a redução de trabalho, de esforços para manutenção e para aumentar a consistência entre tutores; porém deve haver um cuidado pois pode aumentar custos e dificultar a utilização.
- Criação automatizada do conhecimento: capacidade de um tutor inteligente em criar novos conhecimentos a partir de conhecimentos passados pelo autor.

Além destes métodos, outros foram observados por Murray que podem indicar ganhos em usabilidade, como facilidade na edição, flexibilidade na autoria para usuários de diferentes níveis de especialidade, inclusão de mecanismos de busca, cópia e desfazer ações, entre outros.

3.1.2 Paradigmas

Grupos de desenvolvimento de sistemas de autoria costumam visar a construção de ferramentas que permitam o compartilhamento de componentes entre STIs, a redução de custos no desenvolvimento e com foco em paradigmas ou domínios específicos. Por exemplo, o CTAT para sistemas que seguem os paradigmas de modelo de rastreamento

e baseados em exemplos e o ASPIRE para sistemas baseados em restrições. Outros paradigmas são agentes de aprendizagem baseados em diálogos, investigações, jogos, entre outros.

Modelos cognitivos geralmente possuem dois tipos: baseado em regras de produção e baseado em restrições. Enquanto modelos baseados em regras capturam o conhecimento envolvido na geração de instruções passo a passo, os modelos baseados em restrições expressam os requisitos que todas as soluções devem satisfazer (4). Mitrovic (34) aponta vantagens na utilização da modelagem de tutores baseada em restrições ¹, que permite maior exploração do estudante e liberdade na exploração de possíveis soluções, além dos casos que o conjunto de possíveis erros do estudante se torna vasto e por vezes intratável, fazendo com que a enumeração destes erros seja demorada. Porém, não é colocado que a modelagem por restrições é superior à modelagem por regras, ao sugerir que um caminho de desenvolvimento de STIs híbridos possa apoiar diferentes tipos de instruções e diferentes tipos de aprendizes.

Kumar et. al. (23) estudam automatizar a construção de tutores baseados em exemplos ² através de comportamentos de estudantes. Visto que, dependendo da resolução do problema e de seus múltiplos caminhos de resolução, a modelagem do grafo do comportamento do estudante pode se tornar muito complexa. A construção de tutores através de exemplo é composto por três fases: (1) desenvolvimento da interface do usuário, (2) demonstração do comportamento e (3) generalização e anotação do grafo de comportamento. As ferramentas CTAT e ASSISTment, embora apoiem os autores nestas fases, todo o processo é feito manualmente, e a proposta de (23) é automatizar a etapa 3 de generalização do grafo de comportamento, através de um algoritmo que gera automaticamente este grafo, e que mantenha características desejáveis de um grafo de comportamento, que são (23): legível (não necessita de uma compreensão profunda e facilmente editável), completo (pelo menos um caminho de solução deve ser incluído), mensurável (livre de erros e detecta os eventos

¹Do inglês *Constraint-Based Modeling* - CBM.

²Do inglês *Example-Tracing Tutors*.

corretos e incorretos do estudante no seu estado de resolução corrente) e robusto (habilidade em se adaptar a diferentes condições de desenvolvimento do problema, identificar caminhos alternativos e suas características).

3.2 CTAT

O “Cognitive Tutor Authoring Tools” (CTAT)³ é um ambiente de autoria que engloba um conjunto de ferramentas para assistir ao autor no projeto, desenvolvimento, na implementação, nos testes e na manutenção de modelos cognitivos (21). Os desafios envolvidos com a modelagem cognitiva incluem (21): análise de tarefas cognitivas e aquisição do conhecimento, programação avançada em inteligência artificial, testes de depuração e extensão e ampliação do conhecimento. O CTAT tem o objetivo de resolver estes desafios, tornar o desenvolvimento mais fácil e rápido para desenvolvedores, possibilitar a construção por educadores que não são especialistas em psicologia cognitiva ou inteligência artificial e permitir fácil conversão desses modelos cognitivos para tutores *model tracing*.

Choksey (10) fez contribuições ao criar ferramentas para o CTAT, além de ter efetuado experimentos que analisaram o tempo de depuração de erros de implementação destes tutores, podendo permitir uma eficiente autoria nas regras de produção (por “eficiência” entende-se o benefício no tempo de implementação e detecção de erros). E apesar de o CTAT apresentar uma boa opção de modelagem para edição dos elementos da memória de trabalho e do comportamento dos estudantes, os autores ainda devem ter um certo conhecimento prévio referente ao JESS para a edição das regras de produção, visto que a edição destas regras é feita através da sintaxe desta ferramenta.

Em Alevén (5) são apresentadas as vantagens do CTAT na autoria e avaliação dos ganhos em eficiência na construção de STIs. Neste estudo, foram verificados ganhos de eficiência na construção de tutores até duas vezes mais rápido se comparado a outros sistemas tutores inteligentes, utilizando a abordagem baseada em exemplos. Foi a primeira

³Disponível em <http://ctat.pact.cs.cmu.edu/>

avaliação com foco na eficiência da autoria relatada na literatura de STIs.

Olsen et. al. (40) desenvolveram uma extensão para o CTAT de abordagem colaborativa ⁴ para o desenvolvimento de STIs colaborativos ao utilizarem múltiplos motores de tutores sincronizados, um para cada estudante. A CSCL procura proporcionar entendimento e explicações ao apoiar tarefas e interações dentro de um grupo, baseados em scripts colaborativos (por exemplo, “consciência cognitiva coletiva”, “compartilhamento de informação única”, “serra de vai-e-vem”⁵ e “paradigma tutor/aprendiz”.) No caso deste estudo de Olsen, foram exemplificados os scripts de “consciência cognitiva coletiva” e “compartilhamento de informação única” em um tutor que ensina frações equivalentes, cujos testes aplicados com estudantes mostraram ganhos na aprendizagem.

3.3 SimStudent

Jarvis (20) aplicou técnicas de aprendizagem de máquina no CTAT para automatizar a geração de regras de produção na construção de STIs, ao utilizar múltiplos exemplos. Como comparação, o sistema de autoria Demonstr8 (8) induziu regras de produção a partir de um único exemplo usando um mecanismo análogo ao ACT-R. A aprendizagem baseada em exemplo é um tipo de Programação por Demonstração (20) apud (13; 26). Tendo estes estudos como base, Matsuda (32) apresenta um agente de aprendizagem de máquina chamado *Simulated Student* que observa a demonstração de soluções feita pelos autores e induz um conjunto de regras de produção que são generalizadas das soluções encontradas.

Mais tarde, Matsuda (33) retorna a abordagem acerca deste agente agora chamado *SimStudent* e integrado ao CTAT que auxilia os autores a desenvolverem tutores cognitivos sem eles precisarem conhecer programação mais pesada, auxiliando na criação do modelo especialista de duas formas. O *SimStudent* efetua a simulação dos passos de um estudante além de solicitar dicas, e o autor presta a tutoria para ajudar o *SimStudent* a responder

⁴Termo em inglês *Computer Supported Collaborative Learning* (CSCL).

⁵Do inglês *jigsaw*.

corretamente às questões. De outra forma, o autor demonstra passos corretos da resolução e o *SimStudent* tenta induzir princípios de domínio ao generalizar os exemplos elaborados. No artigo apresentado (33) é feita uma análise mostrando que a primeira abordagem é mais eficaz que a segunda.

(27) também fizeram uma avaliação do *SimStudent* tendo como base o a abordagem baseada no rastreamento por exemplos. *SimStudent* conseguiu generalizar regras e diminuir o processo de autoria em até 50% se comparado a autoria por rastreamento por exemplos. É uma economia representativa, visto que a abordagem em rastrear por exemplos efetua a autoria de um exercício, enquanto as regras geradas pelo *SimStudent* podem ser reaproveitadas em outros exemplos. Porém algumas super-generalizações do *SimStudent* devem ser reduzidas.

3.4 ASSISTment

ASSISTment (45; 46) é um sistema para ambiente web, cuja finalidade é auxiliar os professores na avaliação e na assistência aos alunos. Inclui ferramentas de autoria que permitem professores e pesquisadores projetar, implementar e efetuar experimentos para ensino de matemática. A motivação principal do ASSISTment é permitir aos autores a criação, a análise e a avaliação de experimentos entre diferentes formas de ensino, permitindo uma avaliação mais detalhada das habilidades dos estudantes para o professor e possibilitar a inferência sobre qual a melhor forma de ensino que irá contribuir para a melhoria do desempenho dos estudantes. Razzaq aponta que STIs no geral têm dificuldade em fornecer dados empíricos sobre a melhor forma de melhorar o desempenho dos estudantes, além de necessitar de habilidades de programação de computadores para a realização destas tarefas, o que muitas vezes acarreta em maiores esforços por parte de professores e pesquisadores a projetar, conduzir e analisar tais experimentos. Também o tempo limitado de aula disponível no ensino médio obriga os professores a terem que escolher ou gerenciar o tempo entre apoiar o desenvolvimento dos alunos ou avaliar a capacidade deles.

Um dos objetivos do ASSISTment é reduzir algumas destas dificuldades. O sistema aproveita as informações passadas pelos alunos e, conforme é utilizado, aprende sobre as habilidades podendo fornecer previsões mais precisas para auxiliar em determinadas tarefas. Espera-se que os professores sejam capazes de usar esse feedback detalhado para adequar a sua instrução para se concentrar nas dificuldades específicas identificadas pelo sistema. Ao contrário de outros sistemas de avaliação (só avaliam), o “Assistment” proporciona aos alunos tutoria inteligente, enquanto a informação está sendo coletada.

3.5 ASTUS

ASTUS é um framework baseado em restrição que visa diminuir esforços na construção de tutores *model-tracing* e possibilitar a customização dos modelos pedagógicos e do aluno para diferentes perspectivas pedagógicas. Paquette (43) efetuou uma comparação entre as ferramentas de autoria ASTUS e CTAT através da implementação de um tutor de ensino de subtração, com vantagens do ASTUS em relação ao suporte ao estudante. Porém ASTUS, além de demandar mais estudos de viabilidade com relação a seus módulos e de comparação com outras ferramentas de mesma natureza (como a ASPIRE também baseada em restrição), ainda necessita de um conhecimento maior de programação para sua manipulação e criação de componentes de conhecimento mais complexos, fato que o CTAT obtém vantagem por proporcionar esta parte mais acessível para usuários que não dominam programação.

Paquette et. al. (42) apresentam uma abordagem para automatizar dicas para o próximo passo de um tutor *model tracing* construído com ASTUS, um framework de autoria. Enquanto as dicas de uma próxima etapa de um tutor são geralmente criadas pelos autores manualmente, esta proposta geram-nas automaticamente através de modelos (*templates*) de textos para tarefas independentes, ou seja, com identificadores envolvidos da meta a ser atingida pela etapa, das submetas e dos procedimentos utilizados para alcançar esta meta. Além destes indicadores, levam-se em conta para a construção de dicas as condições envolvidas da meta a ser atingida, com modelos de textos associados para

cada expressão condicional (*e, ou, existe...*). Os autores também apontam para trabalhos futuros a utilização de técnicas de linguagem natural, visto que utilizam modelos de textos para a geração das regras. A partir desta geração automática de dicas espera-se que os esforços sejam reduzidos especialmente se múltiplas dicas são associadas com uma mesma unidade de conhecimento ou quando estas dicas tenham que ser traduzidas em muitos idiomas.

3.6 RUI

Direne (16) desenvolveu um conjunto de ferramentas para autoria de STIs no ensino de diagnósticos por imagens de radiologia, através da interpretação, análise e exploração destas imagens, denominado *Representations for Understanding Images* (RUI) (16). O sistema RUI possui um módulo de autoria e um módulo de ensino. O módulo de autoria recebe o conhecimento sobre o domínio dos autores. São informados os conceitos e conhecimentos sobre as classes de anomalia e as imagens e suas descrições para formar a base de conhecimento do STI. O módulo de ensino é o responsável pela transmissão para o aprendiz do conhecimento colocado na base de conhecimento do STI pelo especialista e possui as seguintes partes: interface de ensino, interpretador de regras de ensino e interpretador de linguagem natural.

Pimentel (44) desenvolveu uma extensão para o sistema RUI através do desenvolvimento de uma ferramenta de autoria chamada *Sequence*, que permite a seleção da próxima melhor imagem baseada em cálculos das medidas cognitivas. Para atingir este objetivo, foi desenvolvido um estudo acerca de encontrar e descrever medidas de relevância cognitiva para ordenar imagens de uma base específica, possibilitando que o modelo tutorial de um STI possa modificar a ordem de ensino de imagens. As medidas cognitivas encontradas auxiliam na quantificação do potencial que a imagem tem para exercitar o aprendiz em uma etapa de ensino, além de medirem e representarem computacionalmente a carga cognitiva das imagens. Com a avaliação da carga cognitiva das imagens, as medidas podem permitir à individualização do ensino.

3.7 FARMA

A Ferramenta de Autoria para Remediação de Erros com Mobilidade na Aprendizagem (FARMA) é um projeto desenvolvido por Marczal em seu doutorado (31), no Departamento de Informática da Universidade Federal do Paraná. Esta ferramenta permite a construção de exercícios que envolvam expressões aritméticas e/ou algébricas. Marczal procurou desenvolver a ferramenta conforme os estudos de Murray (38), que propôs que as ferramentas de autoria sejam simples com relação à utilização pelos autores, permitindo assim o fácil desenvolvimento de objetos de aprendizagem ao reduzir esforços em sua construção.

Além disso, a FARMA promove a aprendizagem por erros, pois além de permitir ao aprendiz uma retroação ao ponto onde sua estratégia de resolução falhou, permite também ao professor-autor registrar dicas que sejam úteis para a remediação destes erros.

Além da continuidade que a FARMA teve com outras pesquisas, pode-se destacar a pesquisa de Leite (24), que teve como suporte importante em sua pesquisa a parceria com a FARMA para validar sua arquitetura.

3.8 Outros trabalhos de autoria de regras

Zancanaro et. al. (50) abordam um sistema de regras de produção baseado em frames para um jogo educacional chamado *Renaissance*. Duas interfaces implementadas: *Knowledge Base Editor* e *Knowledge Base Shell*. KBE permite manipular graficamente as estruturas dos frames, definir e editá-los, além da edição das regras e sua exportação em formato XML.

Lógica anulável (*defeasible*) pode expressar facilmente conflito entre regras. A lógica anulável não é uma linguagem para o usuário final, somente para desenvolvedores, por isso Bassiliades (7) apresenta uma ferramenta de autoria gráfica de criação de regras que age como um shell para o sistema de raciocínio anulável DR-DEVICE para a Web. É ainda

apresentada uma arquitetura e funcionalidade bem desenhada do DR-DEVICE além da forma como é feita esta tradução das regras. O DR-DEVICE utiliza uma linguagem baseada em CLIPS (assim como o JESS) compatível com RuleML (através de tags como em XML) e semântica orientada a RDF. O shell gráfico desenvolvido busca facilitar o desenvolvimento e invocação das bases de regras. A base de regras é exibida em formato de árvore de regras em XML, que pode ser facilmente editada dentro do quadro do DR-DEVICE, e depurada através de outra janela deste sistema.

3.9 Conclusões

Embora procedimentos e paradigmas para elaboração de uma ferramenta de autoria não chegam a serem padronizados ou de utilidade obrigatória para compor ambientes de autoria em STIs, Murray elencou métodos e práticas para a elaboração destes ambientes de autoria, que possam potencializar a construção de STIs. Planejar um sistema de autoria para STIs deve garantir que o sistema resulte em um STI de fato, por isso é recomendado que os módulos que compõem um STI (49) sejam bem definidos, e esta recomendação pode ser alcançada se a ferramenta de autoria especificar bem estas diferenças e a comunicação entre os módulos.

Murray também procurou em seu trabalho (37) categorizar alguns sistemas de autoria existentes. O objetivo maior no projeto de um ambiente de autoria é que os módulos que compõem a arquitetura de um STI (Interface, Modelo de Domínio, Modelo do Estudante, Modelo de Ensino (49)) tenham suas entradas e saídas bem modeladas e que faça o que o autor espera ao criar o STI. Também é de interesse da área que os sistemas de autoria permitam uma maior flexibilidade na adaptação de seus módulos e conteúdos.

O CTAT mostrou-se um sistema de autoria com várias aplicações, pesquisas e ferramentas integradas, além de análise de eficiência. Por isso, este trabalho de autoria de regras de produção irá aproveitar alguns conceitos empregados nestas pesquisas acerca do CTAT, por utilizarem ACT-R, regras de produção e Programação por Demonstração

(abordagem baseada em exemplo).

A FARMA, além de ser da mesma linha de pesquisa, utilizada e estendida em outros trabalhos como o de Leite, pode possibilitar a integração e adaptação dos módulos da autoria de regras de produção com seus módulos.

Algumas avaliações desses sistemas de autoria foram feitas e indicaram ganhos no tempo e na aprendizagem. Cada módulo da arquitetura de um STI possui diversas metodologias disponíveis para sua composição. Por mais que duas ou mais ferramentas de autoria sejam semelhantes com relação ao domínio, modelo do estudante ou modelo pedagógico, por exemplo, podem diferenciar na metodologia ou teorias cognitivas empregadas. Por isso que os testes avaliativos enfatizam em seus estudos uma ou outra abordagem separadamente, de um modelo específico da ferramenta.

CAPÍTULO 4

ARQUITETURA DE REMEDIAÇÃO DE ERROS COM MÚLTIPLAS REPRESENTAÇÕES EXTERNAS PROPOSTA POR LEITE

Este capítulo tem a finalidade de descrever a arquitetura de remediação de erros matemáticos através das Múltiplas Representações Externas proposta por Leite (25) (24). A compreensão da arquitetura é essencial para a proposta de autoria que será feita para suas regras de produção. Esta arquitetura visa diagnosticar e remediar o erro apresentado pelo aprendiz por meio de MRE e é constituída de módulos interligados para separar as etapas do processo de remediação por MRE.

As etapas do processo de remediação são as seguintes:

- Detecção do erro do aprendiz;
- Classificação do erro a partir de categorias já definidas para os erros;
- Classificação da função da MRE a partir do tipo de erro;
- Seleção da MRE que auxilie na remediação do erro.

A figura 4.1 apresenta o esquema da arquitetura funcionalista, bem como seus módulos e interligações. A seguir serão apresentados os módulos desta arquitetura.

4.0.1 Módulo Identificador de Expressões

É de responsabilidade deste módulo receber a resposta do aluno e identificar se ocorreu erro ou acerto. Possui como objetivo fazer a conexão entre o OA e o sistema de remediação de erros. Os módulos subsequentes a este somente serão acionados se algum erro for detectado, sendo o Módulo Classificador do Erro acionado primeiramente. Segundo a

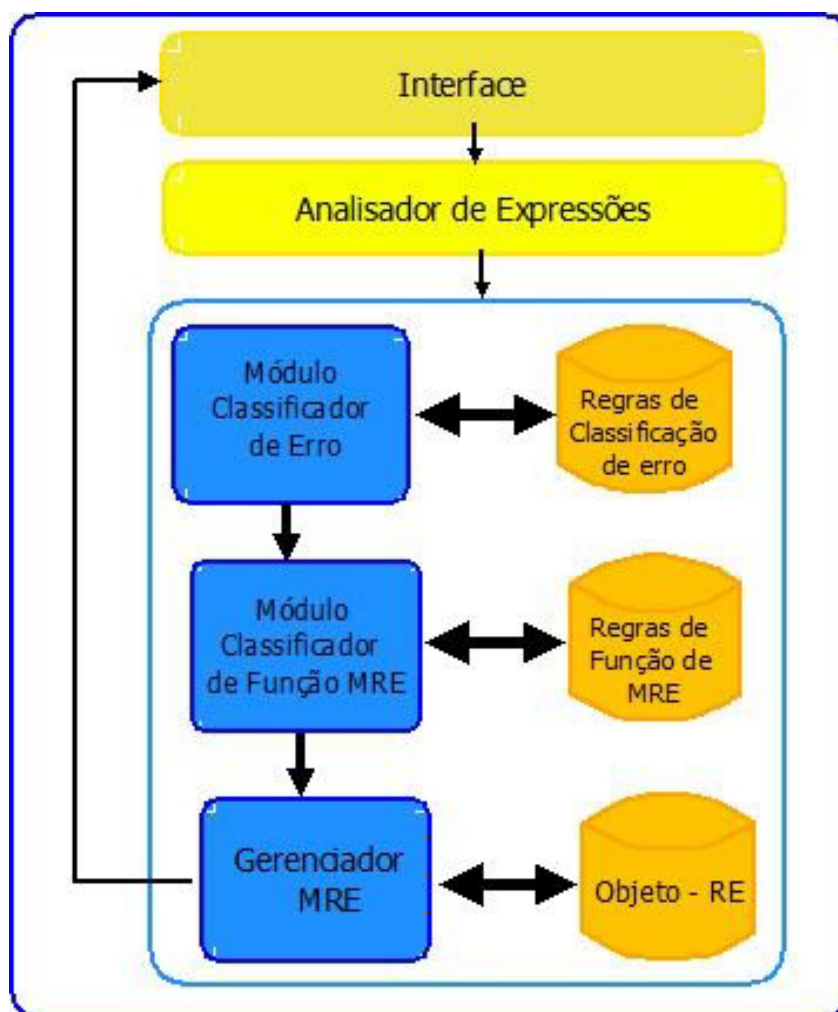


Figura 4.1: Arquitetura Funcionalista de Remediação de Erros com MRE (25).

arquitetura funcionalista, o Módulo Identificador de Expressões irá passar para o Módulo Classificador de Erro as seguintes informações:

- Erro: capturado da interação do aprendiz com o objeto de aprendizagem;
- Ação: referente à etapa do processo de resolução corrente do aprendiz. Esta informação é útil para o tipo de MRE a ser apresentado.
- Tentativas: serve para verificar se a MRE apresentada anteriormente ao aluno foi suficiente para o avanço na lição.

4.0.2 Módulo Classificador de Erro

Após a identificação da informação como erro, o Módulo Classificador de Erro irá identificar o tipo de erro cometido. A classificação utilizará a Base de Regras Sobre Classificação de Erros. No estudo de Leite foi criada uma categorização de erros embasada em estudos focados em erros matemáticos, objetivando a complementariedade de uma classificação única e abrangente. A classificação da autora consiste nos seguintes tipos de erro:

- Interpretação Equivocada da Linguagem: Esse tipo de erro alertaria para a dificuldade do aluno em avançar na compreensão da estrutura do problema para ser formulada uma estratégia.
- Diretamente Identificáveis: Este tipo de erro pode ser sub-classificado em Erro de Deficiência no Domínio ou Uso Inadequado de Dados; Erro de Deficiência de Regra, Teorema ou Definição; e Erro Referente ao Uso do Operador Lógico.
- Indiretamente Identificáveis: Esta classificação contempla o erro apresentado pela falta de lógica correta, neste caso, poderia ser uma classificação incorreta, uma resposta para uma estratégia incorreta e/ou transformação sem avanço.
- Solução Não-Categorizável: o presente erro tem como objetivo contemplar a inexistência de classificação entre os demais. Neste caso, o erro pode indicar que o aluno esteja em algum nível de imaturidade para um conceito resultando em proposição alternativa à resolução.

Outras classificações de erros podem ser consideradas e incluídas nesta arquitetura, dependendo do domínio e das necessidades de cada OA.

Como entrada este módulo receberá o erro, a ação do estudante e o número de tentativas oriundas do Módulo Identificador de Expressões. A Base de Regras sobre Classificação do Erro irá classificar o erro capturado. Com a finalidade de individualizar a remediação do aluno, a ação irá informar em qual subetapa de um conjunto de etapas de resolução

o estudante se encontra, e o número de tentativas irá identificar sucesso ou fracasso da MRE oferecida para remediação.

Após a classificação do erro, este módulo passará para o módulo seguinte, o Módulo Classificador de Função de MRE, as seguintes informações: erro, tipo de erro, ação e número de tentativas.

4.0.3 Módulo Classificador de Função de MRE

A finalidade deste módulo é identificar qual função da taxonomia proposta por Ainsworth (2) das MREs (Papéis Complementares, Restrição de Interpretação ou Compreensão Mais Aprofundada, vistas na seção 2.3) tem correlação com a classificação de erros. Como entrada o módulo recebe o erro, a ação, o número de tentativas e o tipo de erro vindos do Módulo Classificador de Erro. Em seguida, o tipo de erro será validado na Base de Regras sobre Função das MREs. Leite, no papel de especialista nos exemplos descritos em sua tese, efetuou uma correlação entre os tipos de erros e as funções das MREs apresentada na tabela 4.1.

Por exemplo, se o tipo de erro for “Diretamente Identificável” e seu subtipo for “Deficiência de Regra, Teorema ou Definição” então a melhor Função da MRE que possa tratar este erro seria a “Compreensão Mais Aprofundada”. Feita a identificação da Função da MRE, este módulo encaminha para o Gerenciador de MRE as seguintes informações: erro, ação, número de tentativas e função MRE.

4.0.4 Gerenciador de MRE

O módulo Gerenciador de MRE será responsável por definir qual o tipo de remediação que o aluno precisa para resolver o erro cometido e avançar na resolução do problema. Este módulo recebe como entrada o erro, a ação, número de tentativas e a função MRE, não necessitando mais do tipo de erro que foi utilizado para definir a função MRE.

Tipo de Erro	Subclassificação	Função de MREs	Remediação
Interpretação equivocada	-	Papéis complementares	Propor outras formas de apresentar o problema com a possibilidade de o aprendiz fazer uma releitura a partir de uma simbolização matemática.
Diretamente Identificáveis	Deficiência em relação ao domínio ou uso inadequado de dados	Funções de Restrição de Interpretação	Mostrar que, embora a estratégia possa estar correta, a deficiência se encontra no uso das informações.
	Deficiência de regra, teorema ou definição	Compreensão mais aprofundada	Apresentar a regra ou teorema, com o propósito de o aprendiz reorganizar conceito ou generalizar.
	Deficiência na escolha do operador correto	Compreensão mais aprofundada	Apresentar ao aprendiz que seu equívoco encontra-se na escolha do operador correto.
Indiretamente identificáveis	-	Restrição de interpretação	Mostrar ao aprendiz que a lógica adotada não resulta na solução do problema.
Solução não categorizável	-	Compreensão mais aprofundada	Propor ao aprendiz a revisão de conceitos elementares ou presente na base de domínio.

Tabela 4.1: Composição da Classificação de erro e Função de MRE (24).

O Gerenciador irá escolher a apresentação da MRE no momento da remediação do erro já classificado e armazenar a última representação apresentada para avançar na eficácia junto ao aprendiz. Além disso, dentro deste módulo fica o repositório de objetos para representações externas que seleciona qual a melhor representação externa para o aprendiz, levando em consideração os seguintes critérios:

- Persistência no erro: pode indicar uma insuficiência da representação externa em remediar o aprendiz no processo de aprendizagem que acarreta numa possibilidade de mudança no tipo de MRE.
- Sucesso ou fracasso com a representação externa em situações anteriores: pode considerar critérios como idade, familiaridade, experiência etc.
- Maior ou menor complexidade da situação do momento: tarefas mais simples podem exigir representações externas menos complexas, podendo progredir continuamente para outras mais complexas.
- Máximo ou mínimo de suporte: um aluno que mostre um certo nível de conhecimento do domínio pode se mostrar indiferente em relação a algumas representações externas.

A persistência no erro pode ser vinculada ao número de tentativas e o sucesso ou fracasso com representação externa anterior vinculado à ação do aprendiz, podendo assim serem explorados neste módulo. Já a complexidade e o suporte podem ser associados ao número de MREs disponibilizadas para cada etapa do processo, dependendo da contribuição de um especialista do domínio.

4.1 Conclusões

Leite ainda destaca que a classificação proposta demanda de mais estudos e validações para diferentes conteúdos matemáticos, com a possibilidade do surgimento de outras subcategorias conforme a necessidade do domínio a ser estudado. Porém a autoria de regras

que será vista no capítulo 5 seguirá o que foi proposto nesta arquitetura funcionalista, mas nada impedirá que a autoria seja adaptada a mudanças nos atributos das classificações de erros.

Os módulos da arquitetura funcionalista possuem definições de comportamento bem definidas. Porém Leite deixou aberto um leque de possibilidades para o comportamento do Gerenciador de MREs, pois além da função da MRE como entrada para o módulo, critérios para a seleção da melhor MRE podem ser considerados para uma remediação possivelmente satisfatória. Serão propostos alguns critérios de seleção de MRE para exemplificações da autoria e modelagem do estudo de caso.

CAPÍTULO 5

ARPREM - AUTORIA DAS REGRAS DE PRODUÇÃO

Tendo como base esta arquitetura de remediação de erros de Leite (24) apresentada no capítulo 4 e o objetivo de propor uma ferramenta de autoria para as suas bases de regras, a etapa inicial consistiu na sua implementação e, após esta etapa, a criação de um estudo de caso em um protótipo de OA, a ser visto no capítulo 6.

5.1 Decisões de Implementação

Algumas decisões de ferramentas utilizadas para a implementação foram tomadas. Utilizou-se Java¹ como linguagem de programação para implementação por alguns motivos primordiais:

- É uma linguagem de programação que segue o paradigma orientado a objetos;
- Alguns trabalhos da linha de pesquisa utilizam esta ferramenta (22; 15; 30), com possíveis integrações e extensões como trabalhos futuros;
- A comunidade Java possui uma vasta gama de biblioteca de componentes e arcabouços para diversas finalidades, inclusive para a área científica, com soluções prontas que facilitam boa parte da implementação. Como exemplos de ferramentas: ECJ² para o estudo de algoritmos bioinspirados; JESS³ para regras de produção; e a alternativa ao JESS, o JEOPS (14), utilizado neste trabalho.
- Alta portabilidade, pois seu código será interpretado pela máquina virtual e independe de hardware e sistema operacional;
- Facilidade na documentação, dentro do próprio código;

¹Java Enterprise Edition. Disponível em <http://www.oracle.com/technetwork/java/javase/overview/index.html>

²ECJ - A Java-based Evolutionary Computation Research System. Disponível em <http://cs.gmu.edu/~eclab/projects/ecj/>

³JESS - Java Expert System Shell. Disponível em <http://www.jessrules.com/>

- Familiaridade e experiência do autor deste trabalho com a linguagem.

Optou-se também por utilizar o JEOPS (14)⁴ desenvolvido em Java como motor de inferência para processar as bases de regras de produção dos módulos. Com esta ferramenta é possível analisar e interpretar os dados, além de capturar as saídas dos processamentos efetuados nas bases de regras.

Como ferramenta para criação e gerenciamento do banco de dados optou-se pelo SQLite⁵ por ter uma biblioteca compatível com Java e ser multiplataforma, possível incluir em sistemas embarcados, além de salvar a base em um único arquivo, assim como um arquivo de dados.

5.2 Módulos da Arquitetura de Remediação de Erros

A implementação destes módulos seguiu a mesma modularidade proposta na arquitetura de Leite (25; 24), ou seja, cada módulo de sua arquitetura corresponde a um módulo implementado neste trabalho. Também foi criado um controlador em cada módulo, para apoiar as entradas e saídas de suas respectivas bases de regras, além de proporcionar a separação e a independência de cada módulo.

Aproveitando o paradigma de orientação a objetos, procurou-se estruturar a maior parte da arquitetura como classes e objetos do Java e uma tabela correspondente para cada classe no banco de dados. Sendo assim, a modularização da arquitetura de remediação de erros ficou da seguinte forma: Interface do OA, Módulo Controlador Geral, Módulo Identificador de Expressões, Módulo Classificador do Erro, Módulo Classificador da Função das MREs e Módulo Gerenciador da MRE. Estes módulos serão explicados na sequência.

⁴JEOPS - *The Java Embedded Object Production System*. Disponível em <http://www.di.ufpe.br/jeops/>

⁵Disponível em <https://www.sqlite.org/>

5.2.1 Interface do OA

Módulo responsável por interagir com o aprendiz na resolução do exercício. Este módulo consiste em um protótipo de um objeto de aprendizagem. A exigência da Interface para proporcionar uma remediação adequada é que todos os seus componentes que interajam com a arquitetura de remediação de erros sejam identificados individualmente. Esta interação dos componentes da interface com a arquitetura consiste em informar identificadores e valores dos componentes para acionar as regras de produção.

Uma tela de exemplo pode ser vista na figura 5.1. Consiste no texto do enunciado do problema, campos para preenchimento das respostas, uma área para exibição da MRE e, para efeitos de monitoramento das regras e estudo de caso, uma área para exibir as regras acionadas a cada interação. Todos os campos da Interface relativos ao preenchimento da respostas (entradas informadas pelo estudante) irão funcionar como os fatos da memória de trabalho para as regras de produção, seguindo os princípios do ACT-R vistos na seção 2.5.



Figura 5.1: Exemplo de interface que representa um OA para estudo de caso da Arquitetura e autoria implementadas.

5.2.2 Módulo Controlador Geral

O Módulo Controlador Geral é o responsável pelo controle do fluxo que as informações devem seguir entre os módulos. Controla todas as entradas e saídas dos módulos, acionando seus controladores locais (controlador de cada módulo). Recebe a resposta e

as informações que identificam o estado de resolução corrente, oriundos da Interface do OA e repassa para o controlador do Módulo Identificador de Expressões. Depois efetua o gerenciamento das entradas e saídas de cada módulo da arquitetura de remediação de erros, até o final da remediação. Por último, transfere para a Interface do OA as saídas do processamento da remediação, que são a MRE que deverá ser exibida e a descrição das regras que foram acionadas.

5.2.3 Módulo Identificador de Expressões

Assim como na arquitetura de Leite, este módulo irá avaliar a resposta oriunda da Interface do OA e indicar se a resposta é um erro ou um acerto. As informações obtidas da Interface são: a ação do estudante (que inclui informações de estado no qual ele se encontra, resposta informada, meta a ser solucionada para avançar na resolução do problema e componente da interface), pois as regras deste e dos outros módulos utilizam todas estas informações em suas condições.

Estas informações não são passadas diretamente para este módulo. Primeiramente, as informações passarão da Interface para o Controlador Geral, para detecção da necessidade em acionar o Controlador do Identificador de Expressões. Uma vez acionado o Controlador deste módulo, as informações serão as entradas deste módulo e a resposta será avaliada com as regras presentes. Para isso foram desenvolvidas duas bases de regras separadas para este módulo, a Base de Regras das Respostas Corretas e a Base de Regras das Respostas Erradas. O controlador deste módulo irá acionar a Base de Regras das Respostas Corretas e avaliar se a resposta é um acerto.

Caso a resposta seja um acerto, o controlador não aciona a Base de Regras das Respostas Erradas e repassa como saída para o Controlador Geral a informação de que a resposta é um acerto, além da descrição das regras acionadas. Por sua vez, o Controlador Geral não dará continuidade em acionar os módulos da remediação de erro e se encarregará de informar à Interface a próxima etapa de resolução, solicitando a liberação do(s) próximo(s) componente(s) da interface e a exibição da descrição das regras acionadas.

Caso a resposta não tenha acionado nenhuma regra de produção da base das respostas corretas, então a resposta será um erro. O controlador do módulo irá acionar a Base de Regras para Respostas Erradas. Esta base irá verificar se o erro foi previsto pelo autor e que o erro possa ser classificado. Por fim, o controlador deste módulo devolverá a informação de erro para o Controlador Geral poder dar continuidade à remediação, que acionará o Módulo Classificador do Erro.

5.2.4 Módulo Classificador do Erro

Módulo responsável por verificar qual o tipo de erro cometido. O Controlador Geral irá acionar o Controlador do Módulo Classificador do Erro, repassando as mesmas informações obtidas na interface. O controlador local por sua vez efetua a consulta na Base de Regras Sobre Classificação de Erros.

As regras deste módulo retornam os tipos de erros da classificação escolhida pelo autor, lembrando que para estudo de caso será utilizada a classificação de Leite: Interpretação Equivocada, Diretamente Identificável e seus subtipos, Indiretamente Identificável e Solução Não Categorizável. Como saída para o Controlador Geral, o tipo de erro será retornado para ser utilizado no Módulo Classificador da Função da MRE, além da descrição da regra selecionada para exibição na Interface.

5.2.5 Módulo Classificador da Função da MRE

Este módulo ficará responsável pela classificação da Função da MRE a qual o tipo de erro está correlacionado. A correlação entre o tipo de erro e a função da MRE seguiu também a proposta por Leite, a ser vista na figura 6.1 (capítulo 6). O Controlador Geral, além de acionar o controlador deste módulo, passará como entrada as informações da ação do estudante e o tipo de erro classificado no módulo anterior. A Base de Regras deste módulo não necessita mais da resposta do erro, apenas do tipo do erro e das outras informações da ação do estudante. Como saída, este módulo retornará para o Controlador Geral a função da MRE correlacionada ao tipo, além da descrição da regra selecionada.

5.2.6 Módulo Gerenciador da MRE

O módulo Gerenciador de MRE será responsável por definir qual a melhor MRE que o aluno precisa para resolver o erro cometido em uma determinada etapa e avançar na resolução do problema. Este módulo recebe do Controlador Geral como entrada a ação do estudante na etapa corrente, após avaliação pelas outras bases de regras que verificam se a ação foi identificada como erro, a identificação do tipo do erro e da função da MRE a qual o tipo de erro está vinculado. Esta ação contém informações da meta que o estudante deva cumprir, do número de tentativas de resolução desta meta, da função da MRE e do histórico de ações. Todas as informações necessárias para selecionar a melhor MRE candidata a auxiliar o estudante serão requeridas pelas regras de produção conforme o critério de remediação selecionado pelo autor no processo de autoria.

Cada MRE dentro da base de regras das MREs terá uma ou mais regras para sua seleção, pois cada MRE pode aparecer em mais de uma ação do estudante, compor mais de um tipo (gráfico com textos por exemplo), possuir o comportamento de mais de uma função MRE (segundo Ainsworth (2)) ou estar envolvida em um ou mais critérios de seleção. O critério de persistência do erro, por exemplo, terá como entrada o número de tentativas. Já o critério que analisa os sucessos anteriores de remediação terá como entrada o histórico de ações para a verificação das MREs mais promissoras de suporte.

Como saída, este módulo informa a MRE de uma Base de MREs que será exibida na interface, seguindo critérios de seleção da MRE mais promissora para atender o estudante, terminando assim o fluxo de remediação da arquitetura.

5.2.6.1 Critérios de Remediação

Alguns critérios de remediação foram definidos com a finalidade de demonstrar e auxiliar a escolha da melhor MRE para uma remediação definida. Além disso, outros critérios foram apontados pelos estudos de Leite (24) e Oliveira (15):

- “MRE específica para o erro”: caso o autor queira que somente uma MRE seja exibida para esta remediação.

- “Não utilizar MRE específica”: dentre as MREs candidatas, excluir uma em especial, que não seja exibida nesta remediação.
- “Persistência no erro”: informar uma MRE específica caso o estudante cometa o erro em uma determinada tentativa (exemplo, exibir uma MRE específica na terceira tentativa de resolução de uma etapa do problema).
- “Alternar entre MREs da Função da MRE selecionada”: caso o estudante cometa o erro em uma etapa de resolução, alternar entre as MREs pertencentes à função da MRE especificada.
- “Complexidade”: caso o estudante persista no erro, alternar entre MREs considerando sua complexidade (partindo para uma MRE com menor complexidade).
- “Sucessos anteriores com a MRE”: retorna a MRE que obteve o melhor histórico de sucesso para remediar erros.

A figura 5.2 mostra o fluxo completo da remediação de erro e uma numeração que indica a ordem em que as informações são repassadas para cada módulo ou base. Nesta figura não é considerada a representação do fluxo do acerto que termina nas primeiras etapas, visto que o foco é a remediação do erro de fato.

5.3 Arquitetura para a Autoria das Regras de Produção

Além da implementação da arquitetura de remediação de erros (24) e de sua adaptação, é apresentada a arquitetura para autoria das regras. Foram implementados os módulos de Criação da Classificação de Erro, Criação de MREs, Criação de Exercício, Controlador de Definição de Regras, Conversor de Regras e Recuperador de Regras, para suportar a proposta de autoria de regras de produção. Tornou-se conveniente criar uma estrutura de simulação completa de criação do exercício com a finalidade de visualizar todo o processo de autoria de um exercício, por isso foram definidas interfaces, além de auxiliar no processo de autoria. A autoria completa consiste basicamente em três grandes etapas: criação de classificadores de erros, criação de MREs e criação do exercício.

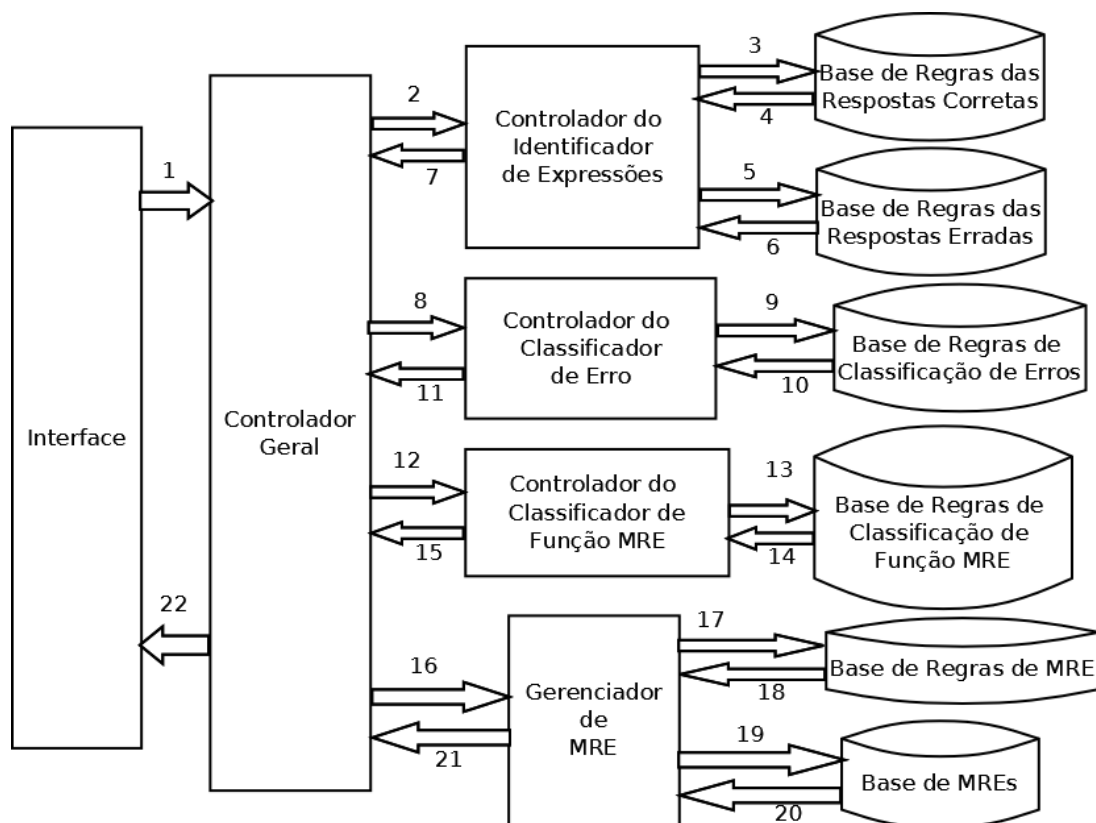


Figura 5.2: Fluxo de informações pela arquitetura de remediação de erros implementada.

5.3.1 Módulo de Criação da Classificação de Erro

Consiste em efetuar relacionamentos de tipos de erros com as funções das MREs. Para efeitos de testes, foi utilizada a classificação de erros e a formatação feita por Leite (25), pois o objetivo deste trabalho não é propor novas classificações de erros, sendo necessários especialistas de domínio para criação destas classificações. O autor irá informar, para cada item do classificador, o Tipo de Erro, Subtipo (se houver), a Função da MRE e uma observação da remediação esperada para este item.

Outros classificadores seguirão este formato (ontologia) de associação de tipos de erros com funções das MREs. Após cadastrado, o classificador será armazenado na Base de Classificadores e utilizado no cadastro da remediação de erro, descrito na seção 5.3.3.3. A figura 5.3 mostra uma proposta de tela para cadastro de um classificador e seus atributos para cada item.

1 - Classificador matemático de Leite ▼

Tipo de Erro	Subtipo do Erro	Função da MRE
Interpretação Equívocada		Papéis Complementares
Diretamente Identificáveis	Deficiência em relação ao domínio ou uso...	Restrição de Interpretação
Diretamente Identificáveis	Deficiência de regra, teorema ou definição	Compreensão mais aprofundada
Diretamente Identificáveis	Deficiência na escolha do operador correto	Compreensão mais aprofundada
Indiretamente identificáveis		Restrição de Interpretação
Solução não categorizável		Compreensão mais aprofundada

Salvar Classificador

Figura 5.3: Parte da tela para cadastro de classificador de erro, a ser utilizado posteriormente nas remediações.

5.3.2 Módulo de Criação de MREs

Módulo responsável pelo cadastro e edição das MREs. Foram definidos alguns atributos com a finalidade de auxiliar nos critérios de escolha da melhor MRE, além de serem baseados nos estudos de Ainsworth (DeFT- (2)). Estes atributos são:

- Complexidade: “1” para MREs menos complexas até “5” para as mais complexas.
- Função ou Funções que a MRE exerce: as três principais da taxonomia de Ainsworth - Papéis Complementares, Restrição de Interpretação e Compreensão Mais Aprofundada (1).
- Formatos que compõem a MRE: foram escolhidos seis formatos - Texto, Equação, Tabela, Imagem, Vídeo e Gráfico.

As MREs e suas informações serão armazenadas na Base de MREs para posterior utilização destes dados, na autoria das regras de produção e no Gerenciador de MREs. A figura 5.4 apresenta a tela de cadastro de uma MRE seguindo a proposta de atributos acima.

5.3.3 Módulo de Criação de Exercícios

Este módulo é o principal da arquitetura da autoria, pois dispõe da interface para o autor interagir com as regras de produção de fato. Foi definida a divisão deste módulo

CADASTRO DE MÚLTIPLA REPRESENTAÇÃO EXTERNA

Descrição

Complexidade

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

Função/Funções

☐ Papéis Complementares

☐ Restrição de Informação

☐ Compreensão Mais Aprofundada

Tipo(s)

☐ TEXTO ☐ EQUAÇÃO ☐ TABELA

☐ IMAGEM ☐ VÍDEO ☐ GRÁFICO

Arquivo:

Visualização

Figura 5.4: Tela de cadastro de Múltipla Representação Externa.

em duas partes: criação do caminho de resolução e criação das remediações.

5.3.3.1 Definição da estrutura de resolução de um exercício

Neste trabalho verificou-se a necessidade de definir a estrutura e como é feita a resolução de um exercício, para auxiliar no estudo de caso e como apoio na construção da autoria das regras de produção.

Um **Exercício** é composto por algumas **Questões**. Uma **Questão** possui um **enunciado** e **campos** a serem preenchidos pelo estudante. Além disso a **Questão** possui um ou mais **Caminhos de Resolução**, que procuram mapear os passos esperados pelo estudante para compreensão do domínio a ser estudado até o final da resolução do Exercício. Cada **passo** de um Caminho de Resolução por sua vez possui uma **Meta** a ser atendida para prosseguir para o próximo passo. Uma **Meta** representa o objetivo que o estudante deve alcançar e possui informações sobre a **resposta** esperada pelo aprendiz, **componente da interface do OA** envolvido que deva receber a resposta do estudante,

meta anterior, próxima meta (pertencente ao próximo passo do caminho de resolução) e indicador booleano mostrando se foi atendida ou não.

Também cada passo do Caminho de Resolução pode possuir **Remediações** para dar suporte ao estudante nos casos de erros. Uma **Remediação**, dentro de um passo de caminho de resolução, irá registrar os possíveis erros que o estudante possa cometer, além de classificá-lo através dos classificadores de erros disponíveis, indicar o tipo de erro, função da MRE e critério para remediação.

A figura 5.5 apresenta um diagrama demonstrando um caminho de resolução, partindo de um estado inicial seguido de três passos compostos cada um por uma meta a ser atingida para avançar para o próximo passo. Cada meta possui suas remediações para erros esperados pelo aprendiz.

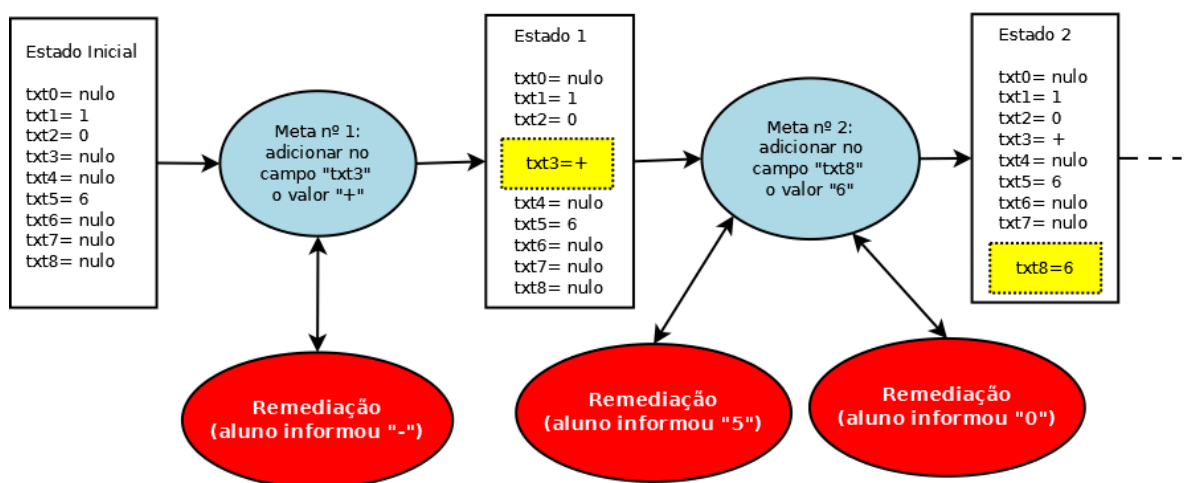


Figura 5.5: Parte de um exemplo de caminho de resolução esperado pelo aprendiz.

O estudante irá efetuar a interação com o Exercício através de **Ações** no objeto de aprendizagem. Uma **Ação** do estudante representa sua resposta informada, além de indicar qual etapa de resolução do problema atual (meta a ser alcançada) e o número da tentativa de resolução desta etapa. Além disso, a Ação irá registrar se houve erro ou acerto e, em caso de erro, qual a medida tomada para a remediação, além da MRE retornada. Esta coleta de dados da ação do estudante é importante para auxiliar o Módulo

Gerenciador de MRE na escolha da melhor MRE.

5.3.3.2 Criação de Caminhos de Resolução

Na tela de criação de caminhos de resolução, o autor cria o exercício com suas questões por demonstração dos passos que o estudante deve seguir para resolução do problema, seguindo os conceitos do ACT (*model tracing*) e Programação por Demonstração (“baseado em exemplos”), também utilizado pelo SimStudent do CTAT (32). Em cada passo é informado o estado dos componentes (quais valores eles possuem no momento) e a meta para atingir o outro estado, que consiste no preenchimento de um valor esperado para um componente. Para cada meta, o autor pode acrescentar as remediações de erro necessárias com a finalidade de auxiliar o estudante no caminho de resolução.

Através do Conversor de Regras, cada meta irá gerar regras de produção para o módulo Identificador de Expressões, assim a arquitetura de remediação de erros identificará quais as respostas corretas, que são as respostas esperadas para cada meta. Para a construção das regras, o Conversor de Regras utilizará informações de identificação do exercício, da questão, do caminho de resolução, da meta, do componente e do valor informado do componente para gerar uma regra identificadora de resposta correta.

A figura 5.6 mostra a interface de autoria para a criação do caminho de resolução de uma questão. Ela apresenta uma parte para o autor informar o enunciado do exercício e modelar o caminho de resolução, e uma parte para visualizar os estados e metas criadas graficamente. Além disso, nesta parte gráfica o autor irá informar suas remediações e efetuar a chamada para o cadastro destas.

5.3.3.3 Criação da Remediação

A tela de criação da remediação é a interface que traduz as informações passadas pelo autor para as regras de produção responsáveis pela remediação do erro matemático. Esta interface é orientada por menus que se adaptam conforme as seleções feitas pelo

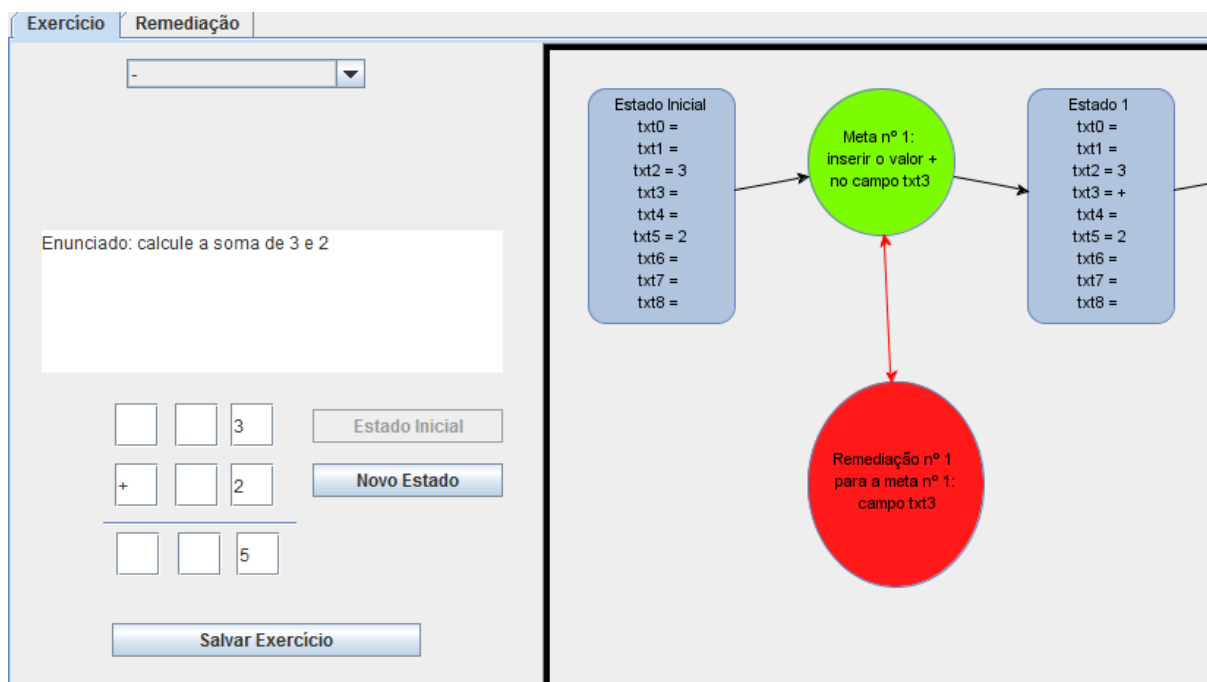


Figura 5.6: Interface de autoria para o caminho de resolução de uma questão do exercício.

autor, através do **Controlador de Definição**. A escolha desta abordagem de interface orientada a menus se dá pelo fato de eliminar equívocos no preenchimento por parte do autor e efetuar todo o processo de categorização do erro. García (18) apud (11) menciona vantagens na abordagem por menus, pois “se bem projetadas são intuitivas, uma forma natural de interação pelo fato de o diálogo humano ocorrer de forma semelhante e menus tornam opções válidas aparentes, minimizando erros e eliminando a necessidade da formulação de comandos e especificação de argumentos”.

O Controlador de Definição auxilia o preenchimento das informações passadas pelo autor, para garantir que a remediação criada consiga retornar uma MRE através de suas regras de produção, se estas regras forem acionadas no OA nas interações dos estudantes. Além disso, monitora as informações para que correspondam ao classificador selecionado para a remediação, cadastrado anteriormente. Assim sendo, o usuário seleciona um classificador dentre os classificadores disponíveis, e o Controlador de Definição selecionará os Tipos de Erros deste classificador.

Após a seleção do Tipo de Erro, o Controlador de Definição irá mostrar os Subtipos de Erro disponíveis para o Tipo de Erro, se houver, para ser também selecionado. Em seguida, após a seleção do Subtipo de Erro, o Controlador de Definição exibirá ao autor qual a Função da MRE correspondente à classificação do Tipo e/ou Subtipo de Erro definido pelo classificador. Esta seleção permitirá ao Controlador de Definição filtrar as MREs candidatas para esta remediação, pertencentes à Função da MRE resultante das seleções informadas anteriormente, garantindo assim uma remediação que possibilite o retorno de alguma MRE. Caso não haja MREs que possuam a Função da MRE selecionada, o Controlador de Definição não permitirá criar regras de produção para a remediação corrente, pois será considerada uma remediação inválida.

A figura 5.7 mostra a tela de remediação de erro que o autor deverá preencher para a criação da remediação. Através das informações desta tela, de forma que o Módulo Controlador de Definição confirme que as regras de produção geradas pela remediação retornem MREs caso sejam acionadas, o **Conversor de Regras** irá efetuar o mapeamento destas informações para as bases de regras de produção.

Cada base de regras de produção da arquitetura de remediação de erros, com exceção da Base de Regras para Respostas Corretas, solicitará um conjunto de informações específicas para sua composição. Para a Base de Regras para Respostas Erradas e Base de Regras para Classificador de Erro, o módulo solicitará informações do exercício, da questão, do caminho de resolução, da meta, do componente e do erro. Para a Base de Regras para Classificador de Função de MRE serão solicitadas informações do exercício, da questão, do caminho de resolução, da meta, do componente, do tipo de erro e do subtipo de erro (se houver). Já as entradas para a criação de regras na Base de Regras para MREs dependerão do critério escolhido, como informações do próprio critério de remediação, do conjunto das MREs que possuam a Função de MRE, da MRE (específica para um determinado erro ou a não utilização de uma MRE na etapa atual) e do número de tentativas (para o critério “persistência no erro”).

Parte 1

Exercício Remediação

ADICIONAR REMEDIAÇÃO DE ERRO

EXERCÍCIO: 1

CAMINHO DE RESOLUÇÃO:

META: nº X -> adicionar no campo "btxY" o valor "Z"

Erro relatado:

Classificador de Erro: 1-Classificador matemático de Leite

Se **Especificar resposta**

Estudante informou uma resposta específica

Então **Classifique o Tipo de Erro como**

1-Interpretação Equivocada

E **Classifique o Subtipo de Erro como**

-

E **Com este tipo de erro**
o Tipo de Função da MRE será

1 - Papéis Complementares

Remediação esperada

Propor outras formas de apresentar o problema com a possibilidade de o aprendiz fazer uma releitura a partir de uma simbolização matemática.

Parte 2

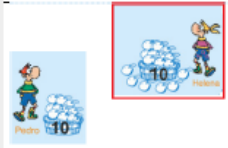
Critério para Remediação 1-MRE específica para o erro

Após número de tentativas:

Especificar MRE:

2-mre 2

Visualização



Salvar Remediação

Figura 5.7: Tela de cadastro de remediação de erro.

A estrutura dos módulos integrados com a arquitetura funcionalista de remediação de erros é apresentada na figura 5.8.

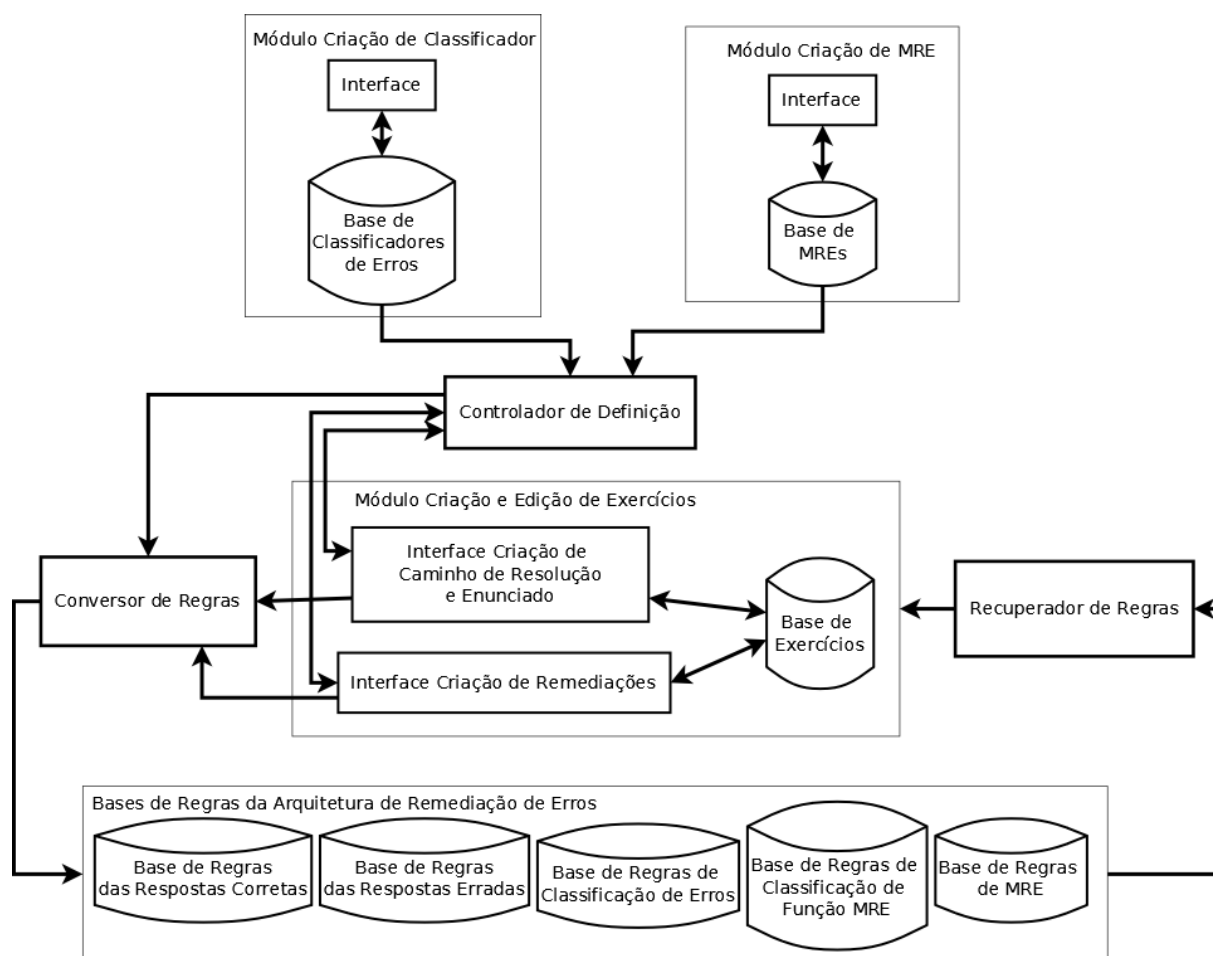


Figura 5.8: Estrutura dos módulos apresentados e suas ligações.

5.4 Esquema para Autoria das Regras de Produção

Uma regra de produção possui uma parte para as declarações da base de conhecimento, uma para a avaliação das condições responsáveis para acionar a regra e outra parte para as ações que modificam esta base de conhecimento. Além disso, a transcrição da estrutura preenchida para a base de regras correspondente possui um mesmo procedimento de tradução para todas as bases de regras.

O Módulo Conversor de Regras possui uma estrutura chamada “**Informações da Regra**” que reproduz o corpo de uma regra de produção. Esta estrutura é utilizada por este módulo para traduzir os parâmetros de criação da regra para a sintaxe do motor de inferência utilizado neste trabalho, antes da edição do arquivo de regras. Esta estrutura possui os seguintes argumentos:

- “**Nome da Regra**”: receberá o nome da regra de produção;
- “**Declarações**”: listagem das declarações da regra;
- “**Condições**”: listagem das condições da regra;
- “**Ações**”: listagem das ações da regra.

Estruturas de três regras de produção são apresentadas na figura 5.9 como exemplo. As regras exemplificadas pertencem às bases de regras para Respostas Erradas, Classificador de Erro e Classificador de Função MRE, respectivamente. Os pontos de interrogação indicam as informações que serão abstraídas da interface de remediação.

Uma base de regras difere de outras bases no que diz respeito às informações adicionadas em suas declarações, condições ou ações, pois cada grupo de regras possui um tipo de saída e objetivos específicos que alteram algum item da base de conhecimento que possa ser diferente de outra base de regras. Por isso, as regras pertencentes a uma mesma base de regras terão sempre a mesma estrutura. A seguir, será apresentada a estrutura de cada grupo de base de regras de produção da arquitetura de remediação de erros.

Regra RespostaErrada_Exercicio_? _Questao_?_Caminho_?_Meta_? _Componente_?_Remediacao_? _Indice_?:	Regra ClassificadorDeErro_TipoErro_? _SubtipoErro_?_Exercicio_?_Questao_? _Caminho_?_Meta_?_Componente_? _Remediacao_?_Indice_?:	Regra ClassificadorDeFunçãoMRE_Exercicio_?_Questao_? _Caminho_?_Meta?_Componente_?_Remediacao_? _Indice_?:
declarações	declarações	declarações
exercício	exercício	exercício
questão	questão	questão
caminho	caminho	caminho
meta	meta	meta
componente	componente	componente
condições	condições	condições
exercício = ?	exercício = ?	exercício = ?
e questão = ?	e questão = ?	e questão = ?
e caminho = ?	e caminho = ?	e caminho = ?
e meta = ?	e meta = ?	e meta = ?
e componente = ?	e componente = ?	e componente = ?
e valor do componente = ?	e valor do componente = ?	e valor do componente = ?
ações	ações	ações
marque resposta como	marque o tipo de erro como ?	marque a função MRE como ?
'errada'	E marque o subtipo de erro	
e resposta errada recebe ? como ?		

Figura 5.9: Exemplos de estruturas das regras de produção para a autoria.

5.4.1 Autoria para as Respostas Corretas

Grupo mais simples para criação das regras, só trabalhando com a Base de Regras sobre Respostas Corretas. A autoria para registro das regras na Base para as Respostas Corretas utiliza informações das metas cadastradas de um caminho de resolução. No momento do cadastro do exercício e a criação dos passos do caminho de resolução que as regras de produção desta base serão geradas. Cada passo do caminho de resolução irá gerar uma regra de produção, utilizando os dados de identificação do caminho de resolução, a identificação da meta pertencente ao passo, a identificação do componente da interface envolvido neste passo e a resposta esperada para indicar que o passo do caminho de resolução foi cumprido e a meta satisfeita.

A figura 5.10 apresenta o esquema de autoria da regra de produção para as respostas corretas, com a estrutura da regra e as variáveis indicando quais partes que as informações serão preenchidas, conforme indicado pelo autor na interface. A variável indicada por “a” será preenchida pela identificação do **exercício**, a variável “b” pela identificação da **questão**, a variável “c” pelo **caminho de resolução**, a variável “d” preenchida pela **meta** deste caminho, a variável “e” preenchida pelo **componente da interface** envolvido no passo e por fim a variável “X” que receberá a **resposta** de fato esperada para a resolução do passo. Como comportamento, caso a regra seja acionada, é o mesmo para todas as regras deste grupo, que é marcar a resposta do estudante como correta e marcar a meta corrente como satisfeita.

O Controlador de Definição é responsável por efetuar o preenchimento da estrutura da regra de produção com as informações da interface e da base de exercícios. Ele também encaminha a estrutura da regra preenchida para o Conversor de Regras, responsável por compilar a regra na base além de buscar da base um **índice** para garantir unicidade à nomenclatura da regra a ser criada e preencher a variável “f” da estrutura com este valor. Este índice é obtido ao verificar a quantidade de regras de produção da base.

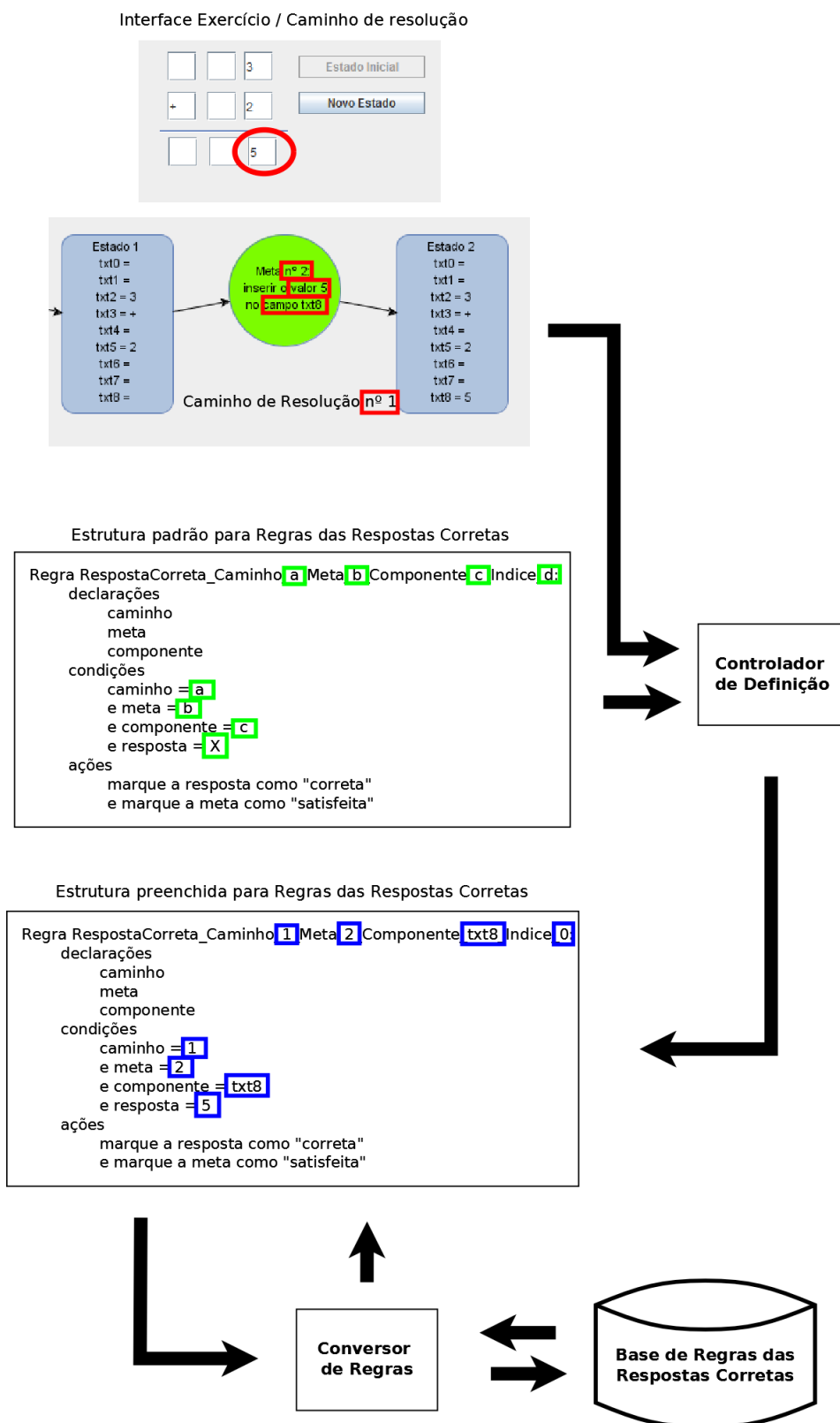


Figura 5.10: Esquema de autoria de uma regra de produção para resposta correta.

5.4.2 Autoria para as Remediações dos Erros

Este módulo é mais complexo que a autoria para as respostas corretas, pois todas as bases de regras, com exceção da Base de Regras das Respostas Corretas, estão envolvidas

nesta etapa. Como entrada, além da meta que foi necessária para as regras das respostas corretas, será informada a remediação para a meta em questão. Esta remediação consiste na informada pelo autor na interface de cadastro de remediação, visto em 5.3.3.3. A remediação é composta pelo erro esperado pelo aprendiz para a meta, tipo de erro, função da MRE mais adequada a tratar este erro e o critério de seleção da melhor MRE que possa ajudar a solucionar este impasse. Cada remediação resultará em uma regra para identificar a resposta errada, uma regra para identificar o tipo de erro, uma regra para identificar a função da MRE e uma ou mais regras para o critério de remediação.

O autor, ao selecionar a meta que queira registrar um possível erro do estudante, será encaminhado à interface de remediação de erros já apresentada na figura 5.7. O Controlador de Definição irá monitorar as informações preenchidas para que a remediação preenchida retorne MREs quando as regras criadas forem utilizadas nas interações com os estudantes. Primeiramente, a tela de remediação, com o auxílio do Controlador de Definição, irá identificar o exercício, bem como o caminho de resolução e a meta a ser remediada. O autor pode descrever o erro que aconteceu, para facilitar possíveis edições. Em seguida, deve selecionar os classificadores disponíveis na Base de Classificadores de Erros para começar a mapear o erro. Na sequência, o autor informará a resposta equivocada para indicar um erro, ou simplesmente informar que ocorreu um erro independente da resposta (resposta diferente da correta e diferente dos erros previstos se for o caso).

Após selecionado o classificador, o Controlador de Definição irá disponibilizar somente os itens de classificação deste classificador selecionado, para o autor poder selecionar o tipo de erro ocorrido. Assim que for definido o tipo de erro, o Controlador de Definição irá mais uma vez atuar na filtragem dos subtipos de erro deste tipo selecionado, se houver. O Controlador de Definição irá apresentar qual a função MRE correspondente ao tipo selecionado pelo autor. Caso o tipo tenha subtipos, a função da MRE só será exibida após a seleção do subtipo pelo autor⁶. Além da função MRE definida, será apresentada a

⁶Toda esta correspondência dos tipos, subtipos e função MRE pertencem ao cadastro do classificador que pode ter sido cadastrado pelo próprio autor ou por terceiros e especialistas do domínio. Esta questão do cadastro de classificadores foi vista em 5.3.1.

descrição da remediação esperada, oriunda também do classificador cadastrado.

Com a função MRE, o Controlador de Definição poderá verificar se existem MREs cadastradas que possuam esta função MRE. Deverá existir pelo menos uma MRE cadastrada que atenda à função MRE selecionada, caso contrário o autor deverá refazer os passos anteriores ou terá que cadastrar ou editar pelo menos uma MRE para que esta remediação seja válida.

Caso esteja tudo preenchido até este ponto, o autor poderá escolher o critério de remediação desejado, também já vistos anteriormente. Dependendo do critério de remediação escolhido, será necessário informar o número de tentativas (para critério de persistência no erro) ou uma MRE de um conjunto (para critério de MRE específica), por exemplo. Mensagens aparecerão caso o autor não preencha a tela corretamente.

Após todos esses passos preenchidos de forma que sejam criadas regras de produção íntegras, o Controlador de Definição fará o preenchimento das estruturas das regras de cada grupo, assim como foi feito na estrutura de regra para as respostas corretas. Nesta etapa de remediação, o Controlador de Definição irá preencher as estruturas das regras para as respostas erradas, das regras para a classificação do erro, das regras para a classificação da função MRE e das regras para o gerenciador de MREs.

Para todas as regras, além do índice para garantir unicidade na nomenclatura da regra, há também a **identificação da remediação** que gerou a regra, para controle no instante de editar ou desativar alguma remediação.

5.4.2.1 Autoria das Regras para Respostas Erradas

A autoria das Regras para Respostas Erradas é muito parecida com a autoria das regras para as Respostas Corretas, só modificando a ação da regra que marca a ação do estudante como errada, além do identificador da remediação na nomenclatura da regra. A figura

5.11 apresenta o esquema de autoria para as regras da resposta errada e um exemplo de preenchimento.

Na primeira parte são identificados quais elementos de entrada da interface serão utilizados para o preenchimento da estrutura da regra. O Controlador de Definição utilizará identificadores do caminho de resolução, da meta, do componente e da resposta para o preenchimento da regra, além de encaminhar a estrutura da regra preenchida para o Conversor de Regras. O Conversor de Regras terá o mesmo comportamento relatado na autoria das regras para respostas corretas, responsável por compilar a regra na base de regras para respostas erradas, além de buscar da base o **índice de unicidade à nomenclatura** da regra criada, preenchendo a variável “g” da estrutura com este valor.

5.4.2.2 Autoria das Regras para Classificador do Erro

Para a autoria das regras do classificador de erro, o Controlador de Definição também utiliza os identificadores de meta, do caminho de resolução, do campo da interface envolvido, da remediação e do valor do erro. Esta estrutura possui variáveis na parte das ações das regras, que receberão o identificador do **tipo de erro** na variável “Y” e **subtipo do erro** na variável “Y1”, selecionado pelo autor na interface, dentre as opções disponíveis para o classificador escolhido, como pode ser visto na figura 5.12. Caso o tipo de erro selecionado não tenha subtipos, conforme indicado por um asterisco (*) na estrutura da regra, o Controlador de Definição não incluirá a ação da regra que classifica o subtipo de erro.

5.4.2.3 Autoria das Regras para Classificador da Função MRE

Além das mesmas informações extraídas da interface de remediação utilizadas pela autoria das regras para classificador do erro, que incluem identificadores de meta, do componente, da remediação, do caminho de resolução, do tipo e do subtipo de erro, com exceção do valor do erro que não será mais necessário para esta etapa, entra a identificação da Função MRE para o tipo de erro escolhido nesta autoria. A informação da Função

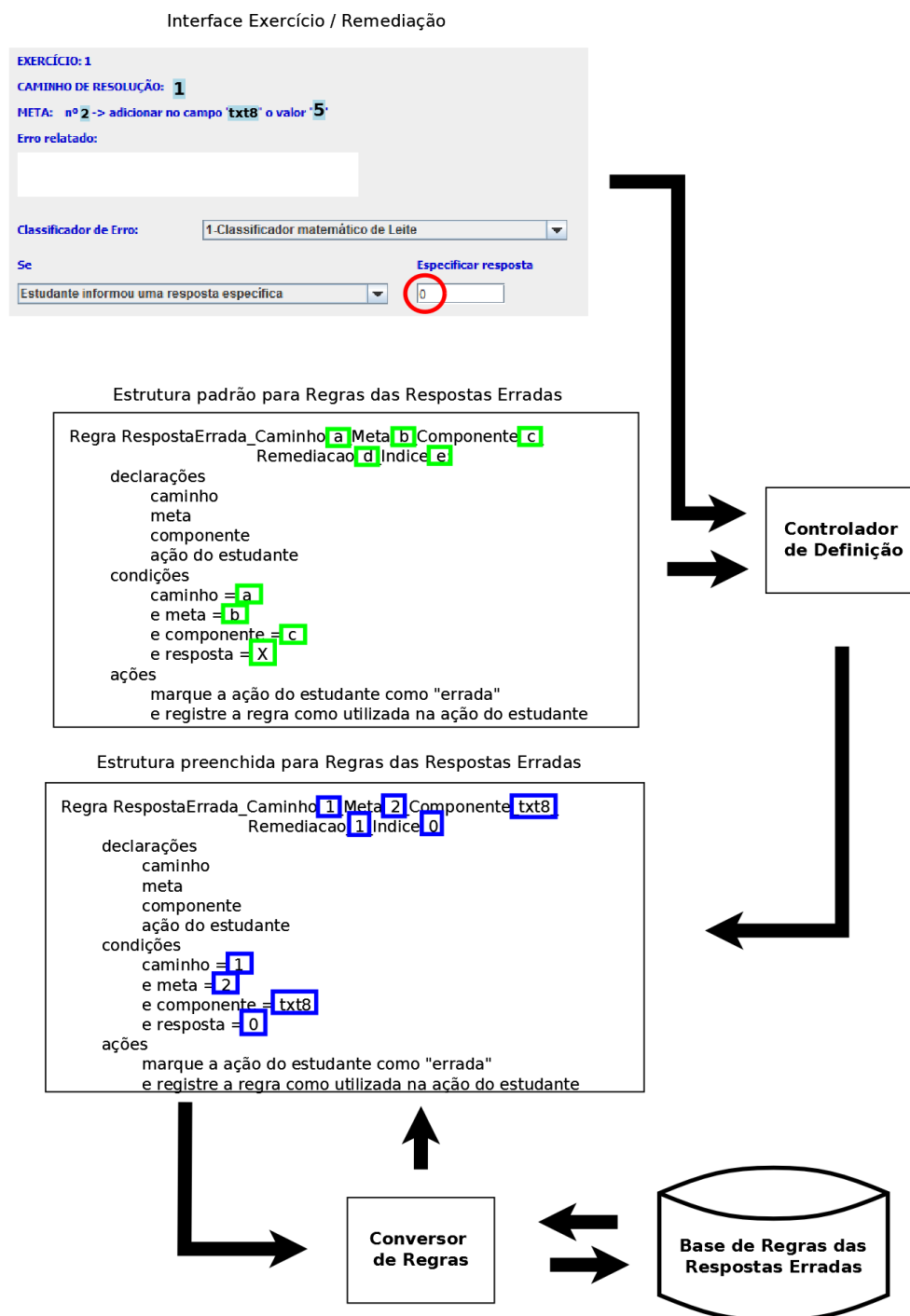


Figura 5.11: Esquema de autoria de uma regra de produção para resposta errada.

MRE já foi mapeada juntamente com o classificador criado, a interligando com um tipo de erro, como pôde ser visto no início desta seção (5.4.2). Esta informação da Função MRE será adicionada na parte de ações da regra para que haja de fato a classificação da função MRE no momento que a regra for acionada.

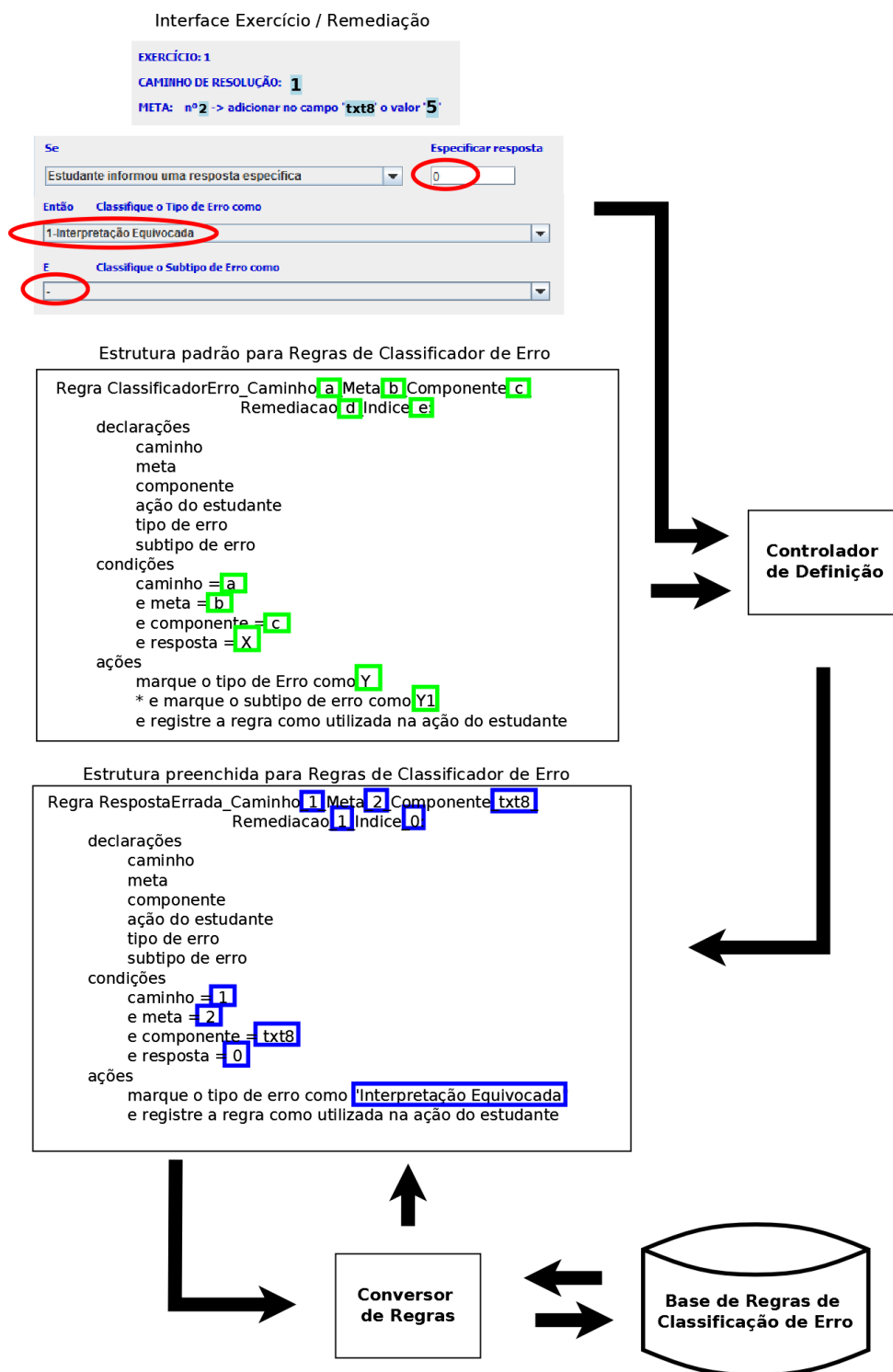


Figura 5.12: Esquema de autoria de uma regra de produção para Classificador do Erro.

A figura 5.13 apresenta o esquema de autoria, assim como nos esquemas já apresentados, a indicação dos elementos utilizados da interface. As linhas do esquema indicados por um asterisco (*) nas partes de declarações da regra e condições da regra indicam que, caso não tenha subtipo de erro para o tipo de erro apresentado, estas informações serão retiradas

da estrutura preenchida, conforme visto neste exemplo. O Controlador de Definição é o responsável por estas omissões.

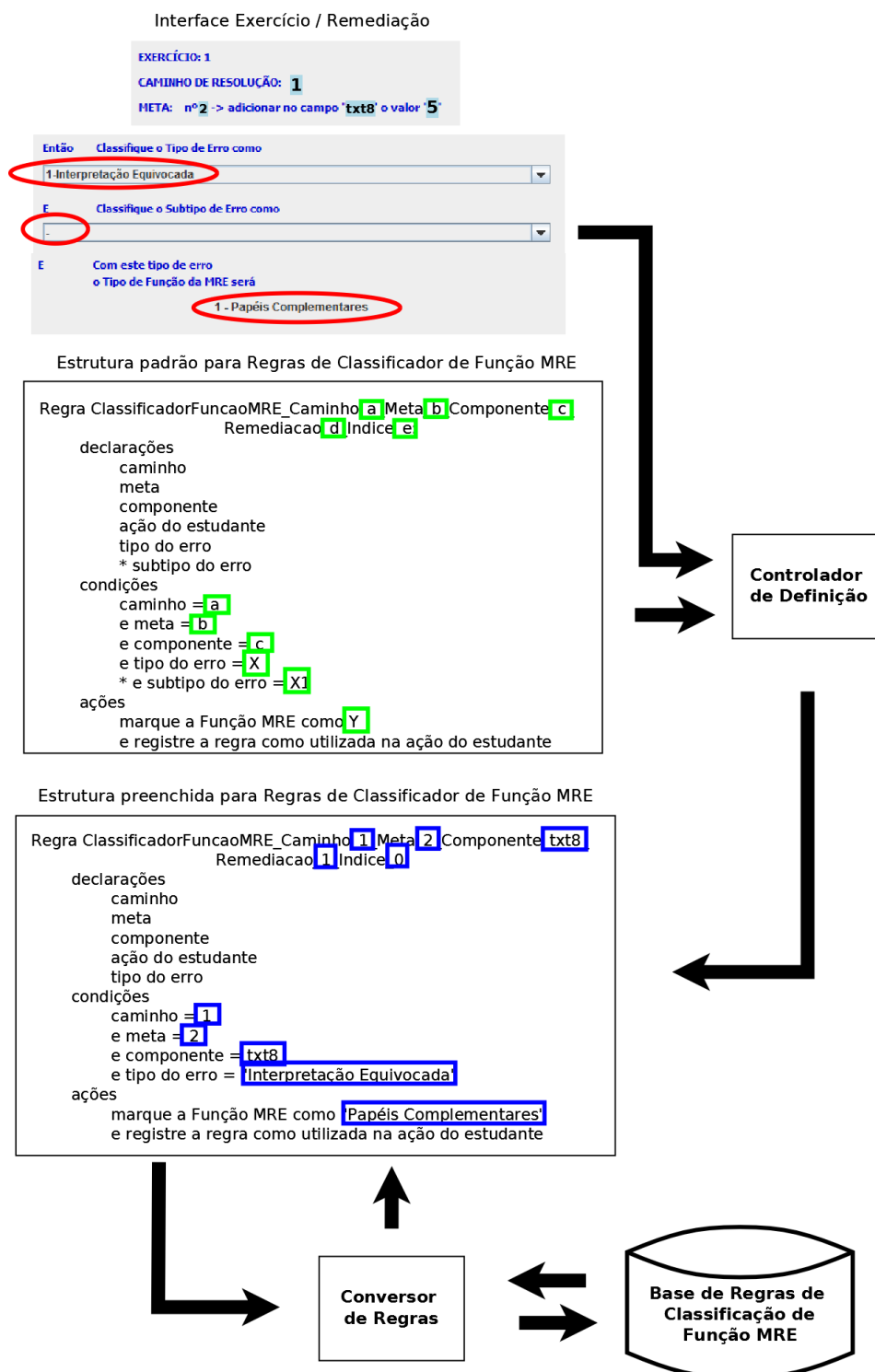


Figura 5.13: Esquema de autoria de uma regra de produção para Classificador da Função MRE.

5.4.3 Autoria para Gerenciador de MRE

Este grupo atende ao último esquema de autoria. É um pouco mais complexo pois envolve os critérios para remediação da melhor MRE candidata à remediação. Por isso, há uma estrutura padrão de regras para cada critério de remediação, pois cada um possui suas próprias necessidades estruturais, apesar de retornarem o objetivo final que é uma MRE.

Com a Função MRE selecionada pelo Controlador de Definição, este também irá listar todas as MREs que possuem esta função MRE, da base de MREs. Estas MREs são MREs candidatas à remediação do erro, por possuírem a Função MRE necessária para o tipo de erro classificado e mapeado pelo autor. Nesse momento entra o critério de remediação que auxilia na escolha da melhor MRE. O próprio autor deve definir qual critério irá atender a remediação.

A figura 5.14 mostra a parte da interface de remediação responsável por informar os dados do critério de remediação. O primeiro campo efetua a seleção do critério de remediação. O segundo campo serve para informar uma numeração para o número de tentativas, caso o critério escolhido pelo autor seja de persistência no erro. O terceiro campo serve para especificar uma MRE caso o critério exija (no caso do trabalho os critérios de “MRE específica” e “persistência no erro”), e uma regra será gerada para seleção desta MRE. Nos outros critérios (“não utilizar MRE específica”, “alternância de MREs”, “complexidade” e “sucessos anteriores”) serão geradas regras para todas as MREs candidatas, com exceção do critério “não utilizar MRE específica”, que vai gerar regras para todas as MREs candidatas que não sejam a MRE selecionada.

As informações para o preenchimento das estruturas diferem em alguns pontos das estruturas apresentadas até agora. Por exemplo, foram omitidos a priori os identificadores do exercício, como questão, meta, caminho de resolução ou componente. Porém, não é descartada a possibilidade em adicionar estas informações futuramente, em bases e exemplos mais complexos. Para a nomenclatura das regras, são utilizados os identificadores da

E Com este tipo de erro
o Tipo de Função da MRE será

1 - Papéis Complementares

Critério para Remediação 1-MRE específica para o erro

Após número de tentativas:

Especificar MRE:

2-mre 2

Visualização

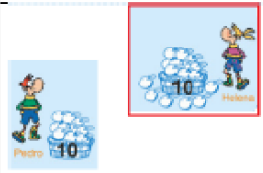


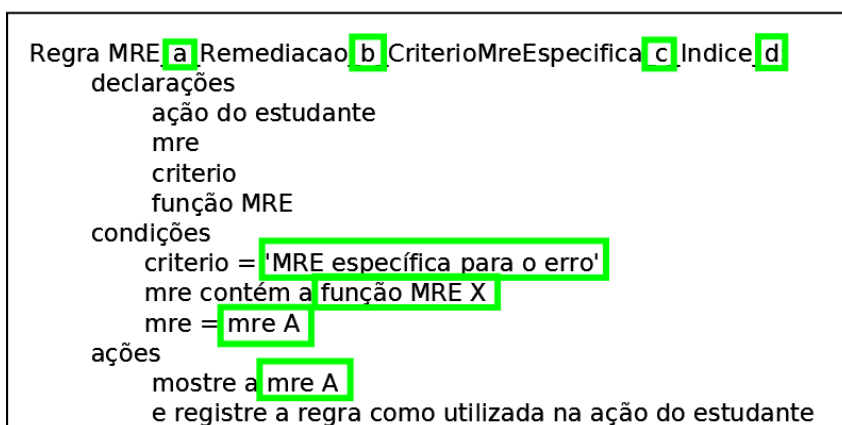
Figura 5.14: Parte da interface de autoria responsável pela interação do autor com a escolha do critério de remediação.

MRE (variável “a”), da remediação (variável “b”), do **critério** de seleção (variável “c”) e do **índice para unicidade** da regra (variável “d”). A identificação da remediação no nome da regra possui a mesma função das regras apresentadas anteriormente, para identificar de qual remediação a regra foi gerada, para facilitar a edição. Além da nomenclatura da regra, o corpo de cada estrutura utiliza os identificadores da **MRE** (variáveis “A”) e da **Função MRE** (variável “X”).

Como o objetivo deste módulo é retornar uma MRE, para todas as estruturas a ação das regras terá o retorno da MRE para o aprendiz, além do registro da regra como utilizada na ação do estudante.

A figura 5.15 apresenta a estrutura dos critérios “MRE Específica” e “Persistência no Erro”, os critérios que o autor deve especificar uma MRE. Na estrutura do critério “Persistência no erro” é utilizado, além das variáveis já relatadas, a variável “N” que indica o **número de tentativas** de resolução que o estudante deve efetuar para aparecer a MRE especificada.

Estrutura padrão para Regras do Critério "MRE específica"



Estrutura padrão para Regras do Critério "Persistência do Erro"

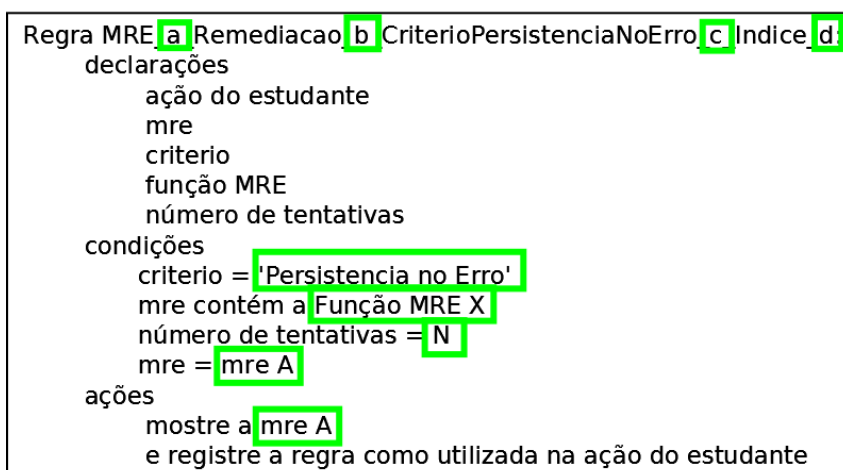
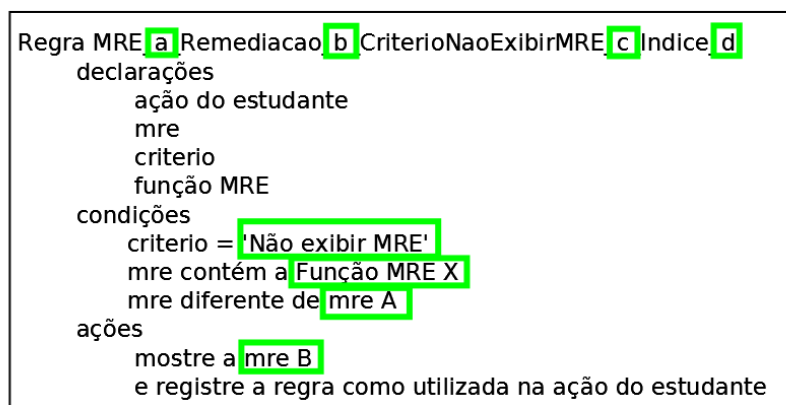


Figura 5.15: Estrutura das regras para os critérios de remediação “MRE Específica” e “Persistência no Erro”.

A figura 5.16 mostra a estrutura dos critérios “Não Exibir MRE Específica” e “Alternar Entre MREs”. Para o critério “Não Exibir MRE Específica”, é incluída a variável “B” que representa cada MRE candidata que pode ser exibida, com exceção da MRE identificada pela variável “A”. Para o critério “Alternar Entre MREs”, também para todas as MREs candidatas, sem exceção, serão geradas regras de produção. Além disso, esta estrutura tem em sua declaração a ação anterior do estudante para garantir a alternância entre as MREs da função MRE em questão.

Por fim, a figura 5.17 apresenta as estruturas dos critérios “complexidade” e “sucessos anteriores”. Estes critérios também geram regras para todas as MREs candidatas. No

Estrutura padrão para Regras do Critério "Não Exibir MRE Específica"



Estrutura padrão para Regras do Critério "Alternar Entre MREs"

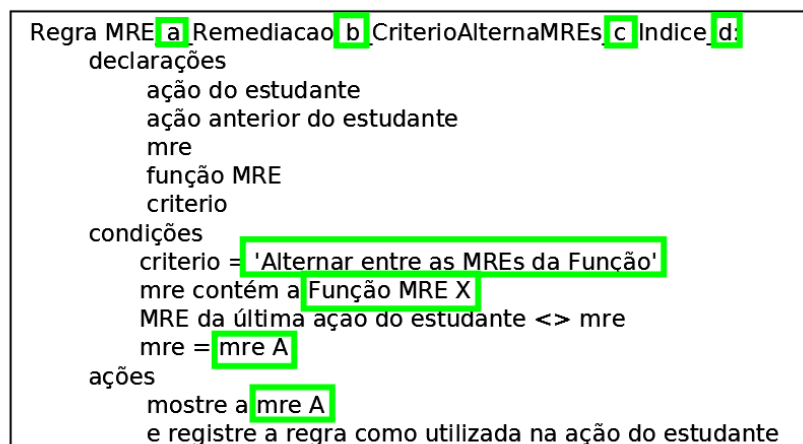


Figura 5.16: Estrutura das regras para os critérios de remediação “Não Exibir MRE Específica” e “Alternar Entre MREs”.

critério “complexidade” é utilizada na declaração a ação anterior do estudante para verificar qual MRE foi utilizada e consultar sua complexidade, para tentar remediar com uma MRE de menor complexidade. Já a estrutura do último critério, que consulta os “Sucessos Anteriores”, apresenta em sua parte da declaração o número de ações satisfatórias da MRE a ser exibida e o número de ações satisfatórias das outras MREs de mesma função, fazendo com que seja exibida a MRE com mais ações satisfatórias.

A autoria das regras do Gerenciador de MRE segue o mesmo esquema que as regras apresentadas anteriormente, com o Controlador de Definição efetuando o preenchimento das variáveis das estruturas e o Conversor de Regras atribuindo o índice no nome das regras para garantir unicidade, além de inserir a estrutura preenchida na Base de Regras

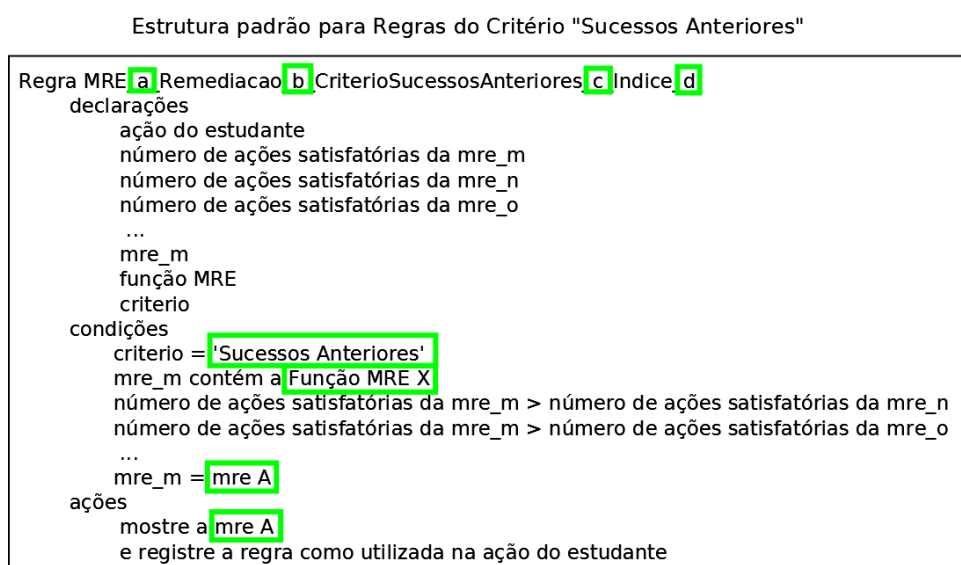
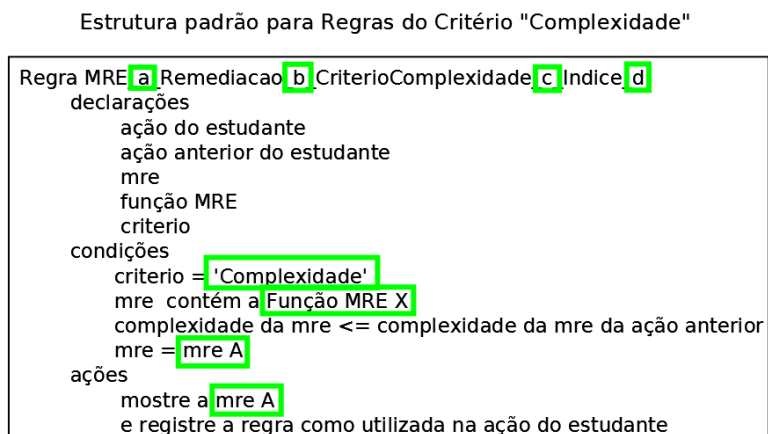


Figura 5.17: Estrutura das regras para os critérios de remediação “complexidade” e “sucessos anteriores”.

de MREs.

Conforme dito na seção 5.2.2, no momento da interação do estudante, o Controlador Geral será responsável por chamar os controladores de cada módulo que possua base de regras e efetuar o empilhamento necessário das informações necessárias que cada base de regras necessita para sua inferência. O Controlador Geral também irá consultar a base de dados das ações do estudante e elencar os dados necessários para o funcionamento, conforme for necessário para o funcionamento da inferência das regras.

5.4.4 Edição e Desativação das Regras de Produção

5.4.4.1 Edição das Regras

A edição das regras possuirá o mesmo processo da criação das regras, com a diferença de que o exercício, o caminho de resolução, as metas e as remediações já estão criadas e devem ser recuperadas da base de dados os quais foram salvos. A interface de autoria permite a edição das regras de produção através das metas (para a base das respostas corretas) e das remediações (respostas erradas), pois é o único contato do autor que não possui conhecimento de programação e edição das regras.

Assim que o autor efetuar a seleção do Exercício, feito através da própria interface que disponibiliza os exercícios cadastrados da Base de Exercícios, a estrutura do caminho de resolução com suas metas e remediações são carregados na interface, bem como o enunciado do exercício. Em seguida, ele pode efetuar edições em algum passo do caminho de resolução, que resultará na edição da resposta correta, ou editar alguma informação da remediação, que resultará na edição da resposta errada esperada.

O módulo **Recuperador de Regras** irá utilizar a identificação da meta para recuperar na Base de Regras para Respostas Corretas a regra relacionada à meta editada e disponibilizá-la dentro da estrutura de regras preenchida. Em seguida, o Controlador de Definição irá substituir os dados editados na estrutura preenchida da regra e chamar o Conversor de Regras para transcrever a regra editada na Base de Regras para Respostas Corretas. Como a arquitetura disponibiliza o registro das ações do estudante, optou-se pela integridade destes registros. Por isso, o Conversor de Regras irá desativar a regra já existente na base de regras e gerar um novo índice para a “nova” regra de produção, antes de inserir na base de regras. O Recuperador de Regras também irá efetuar o mesmo procedimento de leitura das regras nas outras bases de regras caso a meta possua remediações envolvidas, utilizando o identificador de cada remediação para recuperar todas as regras relacionadas.

Caso o autor edite alguma informação de uma remediação, o procedimento é o mesmo, o Recuperador de Regras recupera as regras de todas as bases envolvidas com a remediação do erro (todas com exceção da base de regras para respostas corretas), através do identificador da remediação e inserir cada regra em sua estrutura correspondente. Após o Controlador de Definição atualizar das informações nas respectivas estruturas, o Conversor de Regras irá efetuar o mesmo procedimento de desativar as regras editadas em suas bases e inserir a nova regra com novos índices.

5.4.4.2 Desativação das Regras de Produção

A arquitetura não trabalha com o conceito de exclusão, visto que dados de ações do estudante já ocorridas podem ter utilizado remediações, mas sim no conceito de desativação de uma meta, de um caminho de resolução e, conseqüentemente, da regra de produção. O autor pode desativar uma meta ou uma remediação específica. Caso uma meta seja desativada, todas as suas remediações relacionadas também serão desativadas em cadeia, bem como todas as regras envolvidas tanto com a meta quanto com a remediação.

O Recuperador de Regras efetua a recuperação das regras envolvidas e o Controlador de Definição efetua a desativação das informações na base do exercício, além de habilitar o Conversor de Regras que irá proceder somente em desativar a regra em sua base.

5.5 Conclusões

Este capítulo procurou mostrar e exemplificar todo o processo de autoria de um exercício, através da criação de um exercício e como a arquitetura irá se comportar com a interação com o estudante. Foi apresentado o ARPTEM como arquitetura da autoria das regras de produção como complementar à arquitetura de remediação de erros. O Controlador de Definição das regras é o que acompanha todo este processo de autoria, pois verifica a integridade da remediação e dá garantia que as regras retornem um resultado coerente antes de serem salvas nas bases de regras.

As remediações foram projetadas a nível de cada meta, procurando seguir a teoria do *model tracing* no ACT-R. Desta forma, há o inconveniente de o autor não conseguir inserir uma remediação que seja independente do passo de resolução, que funcione para todo o exercício, sugerindo uma replicação desta remediação para cada meta. Outras estruturas de regras podem ser criadas, que não utilizem informações mais específicas como identificadores de metas.

Conforme já mencionado anteriormente, outros critérios de seleção podem ser propostos e integrados a esta proposta de autoria, bem como a otimização e melhor definição dos critérios inseridos neste trabalho.

A autoria utilizou o procedimento de registrar todo o exercício, suas metas e remediações em paralelo às regras de produção para que o autor possa dessa forma trabalhar indiretamente com a criação e edição dessas regras sem ter conhecimento de programação e de regras de produção, assim os autores não precisam se preocupar com esta parte de programação.

Os dados gerados da arquitetura, como o exercício, os registros dos estudantes e as bases de regras de produção podem ser utilizados e disponibilizados prontos para outros usuários e pesquisadores. O armazenamento das ações do estudante poderá contribuir para uma avaliação qualitativa das remediações, sendo possível trabalho futuro que aproveite esta base de dados, como estudos com *data mining* e aprendizagem de máquina.

Por fim, o Controlador de Definição, aliado à interface orientada por menus, procura efetuar a redução da carga cognitiva na autoria das regras de produção, ao controlar todo o processo, evitando que haja inconsistências na classificação do erro, ou atribuição de MREs de funções MRE diferentes da selecionada.

CAPÍTULO 6

ESTUDO DE CASO

O presente capítulo tem a finalidade de demonstrar através de um exemplo o comportamento de um objeto de aprendizagem utilizando a arquitetura de remediação de erros com regras de produção criadas através da autoria proposta neste trabalho.

6.1 Procedimento de Autoria

Os seguintes cadastros preliminares são necessários para o autor conseguir efetuar a autoria do exercício, caso não existam:

- Cadastro do Classificador;
- Cadastro das Múltiplas Representações Externas.

Para o estudo de caso deste trabalho, o objeto de aprendizagem foi baseado em um exemplo utilizado por Leite para representar a aplicação da remediação no campo da Aritmética (24). Visto que, para fins de estudo de caso da autoria das regras sobre a arquitetura implementada, o exemplo reproduzirá também a classificação proposta no trabalho de Leite, além de aproveitar suas MREs, como pode ser visto na figura 6.1.

Para o cadastro das MREs deste exemplo, será considerado que todas possuam a mesma complexidade, atribuindo o menor valor a elas, pois será adotado apenas o critério de seleção “MRE específica” para todas as remediações cadastradas. Como tipo de MRE, como pode verificar na figura 6.1, todas são do tipo “imagem” com exceção da última MRE que será do tipo “texto”. Além disso, cada MRE será classificada com uma Função MRE correspondente ao que também pode ser verificada na figura 6.1.




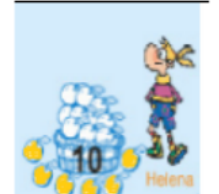

Tipo de Erro	Subtipo	Erro apresentado	Função de MREs	Remediação
Interpretação equivocada	-	O aprendiz não consegue avançar em nenhuma estratégia porque pode estar com dificuldade de passar o problema da linguagem verbal para a matemática.	Papéis complementares	 MRE 1
Diretamente Identificáveis	Deficiência em relação ao domínio ou uso inadequado de dados	O aprendiz não consegue identificar que Helena possui uma quantidade maior.	Funções de Restrição de Interpretação	 MRE 2
	Deficiência de regra, teorema ou definição	aprendiz ainda não se apropriou da parte conceitual, quando não consegue identificar cada personagem com suas respectivas quantidades.	Compreensão mais aprofundada	 MRE 3
	Deficiência na escolha do operador correto	A estratégia do aprendiz apresenta a seguinte solução: $10 - 6 = ?$	Compreensão mais aprofundada	 MRE 4
Indiretamente identificáveis	-	O aprendiz não consegue perceber que a quantidade de laranjas de Helena é maior que a de Pedro.	Restrição de interpretação	 MRE 5
Solução não categorizável	-	Caso o erro cometido não esteja contemplado em nenhuma classificação acima, o erro será incluído nesta seção até ser analisada a necessidade de uma nova categoria.	Compreensão mais aprofundada	Propor ao aprendiz a revisão de conceitos elementares ou presente na base de domínio. Neste caso será apresentado ao aprendiz os principais conceitos aritméticos para retomar a base conceitual do aprendiz. MRE 6

Figura 6.1: Aplicação no campo da Aritmética (24) e as identificações das MREs. A MRE 6 no caso será um texto introdutório à aritmética.

Algumas alterações para a simulação do caso de uso foram feitas, para suportar o conceito de rastreamento dos passos do estudante (*model tracing*), visto que a aplicação original (24) teve como foco demonstrar a categorização do erro matemático, o vínculo

à função MRE e a remediação do erro através de uma MRE, não sendo especificadas as etapas de resolução do problema, somente previstas no seu estudo. O objetivo deste caso de uso é analisar as saídas esperadas pelo autor através de simulação de resolução de uma questão do exercício.

O enunciado do exemplo é o seguinte: *Pedro comprou 10 laranjas e Helena comprou 6 laranjas a mais que Pedro. Quantas laranjas Helena comprou?* Primeiramente, cada campo do exemplo possui uma identificação, conforme a figura 6.2. O autor modela um caminho de resolução esperado, ao preencher primeiramente o estado inicial e indicar os valores para um campo de cada vez, formando novos estados até o estado final desejado pelo autor. Conforme a modelagem do caminho de resolução, o autor pode visualizar a meta para cada passo, que consiste da paridade do campo e o valor que deve ser informado para o avanço da resolução. A figura 6.3 apresenta o caminho de resolução completo.

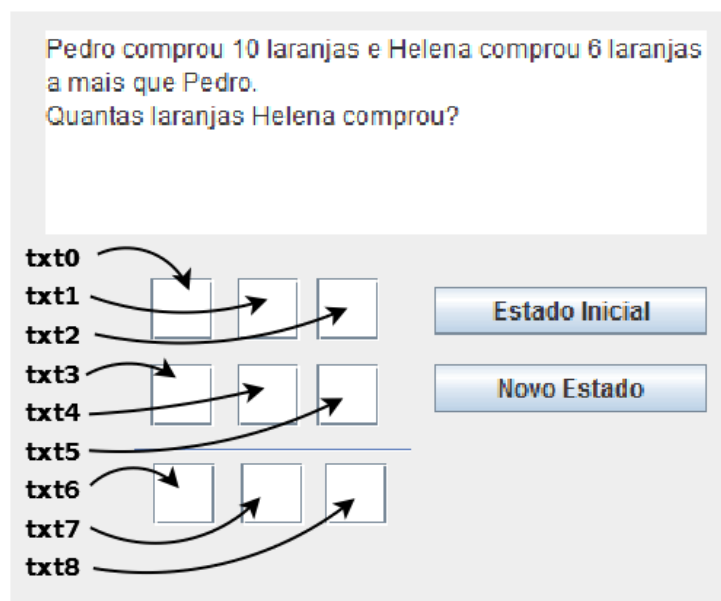


Figura 6.2: Parte da interface de autoria do exercício, com identificação dos campos e enunciado.

Para o estudo de caso, o estado inicial não indica nenhum preenchimento dos campos, e cabe ao estudante iniciar este preenchimento. O autor poderia iniciar o preenchimento de alguns campos, conforme a necessidade e o objetivo do seu exercício. As metas mapeadas no passo de resolução são as seguintes:

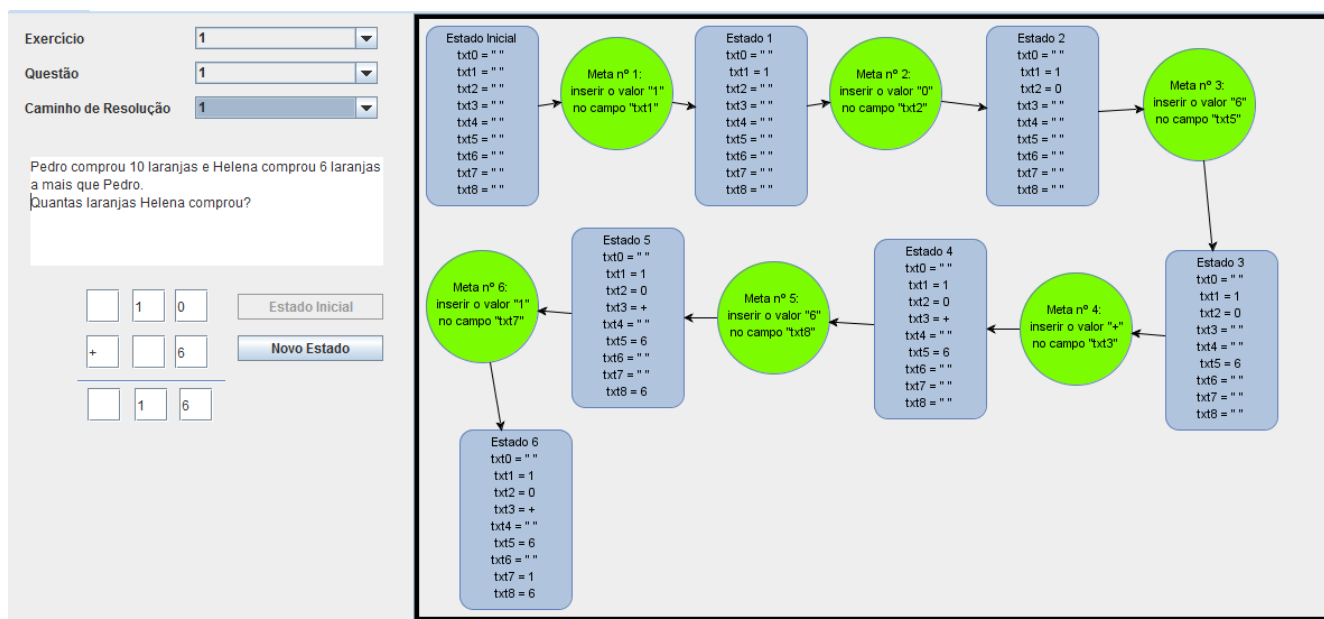


Figura 6.3: Interface de autoria do exercício e a exibição do caminho de resolução com os estados e metas gerados.

- preencher o campo "txt1" com o valor "1" (meta 1);
- preencher o campo "txt2" com o valor "0" (meta 2);
- preencher o campo "txt5" com o valor "6" (meta 2);
- preencher o campo "txt3" com o valor "+" (meta 3);
- preencher o campo "txt8" com o valor "6" (meta 4);
- preencher o campo "txt7" com o valor "1" (meta 5).

Em cada meta é possível gerar remediações, quantas forem necessárias. No caso deste caso de uso foram cadastradas as remediações conforme a tabela 6.1. Após o cadastro das remediações e do caminho de resolução do exercício, as regras de produção foram geradas, conforme esquemas explicados no capítulo 5.

6.2 Análise das Saídas no Objeto de Aprendizagem

Após a criação das regras de produção, verificou-se a necessidade de analisar as saídas retornadas por um objeto de aprendizagem que simula o exercício criado e utiliza

Remediação	Meta	Resposta errada	Tipo de Erro	Subtipo de Erro	Função MRE	MRE
1	meta 1	(resposta em branco)	Interpretação Equivocada		Papéis Complementares	MRE 1
2	meta 2	(resposta em branco)	Interpretação Equivocada		Papéis Complementares	MRE 1
3	meta 3	0	Diretamente Identificáveis	Deficiência em relação ao domínio ou uso inadequado de dados	Restrição de Interpretação	MRE 2
4	meta 1	0	Diretamente Identificáveis	Deficiência em relação ao domínio ou uso inadequado de dados	Restrição de Interpretação	MRE 2
5	meta 1	6	Diretamente Identificáveis	Deficiência em relação ao domínio ou uso inadequado de dados	Restrição de Interpretação	MRE 2
6	meta 2	6	Diretamente Identificáveis	Deficiência em relação ao domínio ou uso inadequado de dados	Restrição de Interpretação	MRE 2
7	meta 3	1	Diretamente Identificáveis	Deficiência em relação ao domínio ou uso inadequado de dados	Restrição de Interpretação	MRE 2
8	meta 4	-	Diretamente Identificáveis	Deficiência na escolha do operador correto	Compreensão mais aprofundada	MRE 4
9	meta 5	4	Diretamente Identificáveis	Deficiência na escolha do operador correto	Compreensão mais aprofundada	MRE 4
10	meta 5	0	Diretamente Identificáveis	Deficiência de regra, teorema ou definição	Compreensão mais aprofundada	MRE 3
11	meta 5	1	Diretamente Identificáveis	Deficiência de regra, teorema ou definição	Compreensão mais aprofundada	MRE 3
12	meta 5	(resposta em branco)	Diretamente Identificáveis	Deficiência de regra, teorema ou definição	Compreensão mais aprofundada	MRE 3
13	meta 6	(resposta em branco)	Indiretamente identificáveis		Restrição de Interpretação	MRE 5
14	meta 6	0	Indiretamente identificáveis		Restrição de Interpretação	MRE 5
15	meta 1	(qualquer outra resposta)	Solução não categorizável		Compreensão mais aprofundada	MRE 6
16	meta 2	(qualquer outra resposta)	Solução não categorizável		Compreensão mais aprofundada	MRE 6
17	meta 3	(qualquer outra resposta)	Solução não categorizável		Compreensão mais aprofundada	MRE 6
18	meta 4	(qualquer outra resposta)	Solução não categorizável		Compreensão mais aprofundada	MRE 6
19	meta 5	(qualquer outra resposta)	Solução não categorizável		Compreensão mais aprofundada	MRE 6
20	meta 6	(qualquer outra resposta)	Solução não categorizável		Compreensão mais aprofundada	MRE 6

Tabela 6.1: Relação das remediações cadastradas para o exemplo.

a arquitetura de remediação de erros com as regras de produção geradas. O objeto de aprendizagem foi elaborado com os componentes necessários para suportar as regras geradas, utilizando os dados do exercício cadastrado, com o enunciado e os campos necessários para entrada do estudante. Além disso, a interface do objeto de aprendizagem possui um campo para exibição da MRE esperada e, para este estudo, um componente para exibição das regras acionadas em cada ação feita no objeto de aprendizagem.

Após a leitura dos dados do exercício da base de exercícios, os campos para interação do estudante com o objeto de aprendizagem estarão orientados pelo caminho de resolução. Ou seja, o campo da meta corrente, que deverá ser preenchido, estará habilitado e os demais bloqueados.

Para analisar o comportamento da arquitetura com as regras de produção foi adotado um procedimento em informar, para cada meta:

- os erros previstos na etapa de autoria das remediações da meta;
- um valor qualquer que não seja o valor esperado pela meta;
- e por último a resposta correta para satisfazer a meta e avançar na resolução do exercício.

O importante deste procedimento é a resposta correta ser informada na última ação de cada meta, pois o acerto ocasiona no avanço da próxima meta e, caso o acerto seja informado primeiro, não permite a listagem das remediações da meta corrente.

Cada ação que ocasionou em uma resposta errada acionou quatro regras de produção da arquitetura, que são uma regra no módulo Interpretador de Expressões, uma no módulo Classificador do Erro, uma no módulo Classificador da Função MRE e outra regra no módulo Gerenciador de MRE. As ações que resultam em resposta correta acionam somente uma regra no módulo Interpretador de Expressões.

Como exemplo, para a meta nº 3, ao ser informado a resposta “0” para o campo “txt5”, o que ocasiona uma resposta errada (“6” seria a resposta esperada para satisfazer a meta), as seguintes regras de produção são acionadas:

----- interação:10

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_3

_COMPONENTE_txt5_REMEDIACAO_3_

Condições:

Resposta = "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_2_suberrortype_1_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_3

_COMPONENTE_txt5_REMEDIACAO_3_

Condições:

Resposta = "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Classifique o tipo de erro como "Diretamente Identificáveis"

e classifique o subtipo do erro como "Deficiência em relação ao domínio ou uso inadequado de dados"

Regra acionada para classificar a Função da MRE:

funcaoMRE_2_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_3
_COMPONENTE_txt5_REMEDIACAO_3_

Condições:

Tipo de Erro = "Diretamente Identificáveis"

Subtipo de Erro = "Deficiência em relação ao domínio ou uso inadequado de dados"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Classifique a Função MRE como "Restrição de Interpretação"

Regra acionada para selecionar a MRE:

regraMRE_2_REMEDIACAO_4_criterion_1_

Condições:

MRE = "MRE 2"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 2"

O mesmo exercício foi executado 30 vezes, cada execução representando um estudante, com 26 interações cada estudante, resultando em 780 interações, para avaliar se as regras de produção retornam as MREs esperadas pelas remediações cadastradas. A tabela 6.2 apresenta as interações ocorridas ao simular a resolução do exercício, com as entradas passadas para cada meta, além das saídas (MREs) para os casos onde não houve acerto. As respostas informadas indicadas por um "X" representam quaisquer respostas que não foram mapeadas na autoria, sejam respostas corretas ou respostas erradas. A última coluna apresenta o número de execuções de cada interação. Ao comparar estas MREs retornadas em cada interação 6.2, percebe-se que as regras geradas pela autoria do exercício, dentro da arquitetura de remediação de erros, se comportaram conforme o esperado na autoria.

Ação	Resposta	Meta	Resposta correta?	MRE	Função MRE	Execuções
1	(em branco)	1	NÃO	MRE 1	Papéis Complementares	30/30
2	0	1	NÃO	MRE 2	Restrição de Interpretação	30/30
3	6	1	NÃO	MRE 2	Restrição de Interpretação	30/30
4	X	1	NÃO	MRE 6	Compreensão mais aprofundada	30/30
5	1	1	SIM			30/30
6	(em branco)	2	NÃO	MRE 1	Papéis Complementares	30/30
7	6	2	NÃO	MRE 2	Restrição de Interpretação	30/30
8	X	2	NÃO	MRE 6	Compreensão mais aprofundada	30/30
9	0	2	SIM			30/30
10	0	3	NÃO	MRE 2	Restrição de Interpretação	30/30
11	1	3	NÃO	MRE 2	Restrição de Interpretação	30/30
12	X	3	NÃO	MRE 6	Compreensão mais aprofundada	30/30
13	6	3	SIM			30/30
14	-	4	NÃO	MRE 4	Compreensão mais aprofundada	30/30
15	X	4	NÃO	MRE 6	Compreensão mais aprofundada	30/30
16	+	4	SIM			30/30
17	(em branco)	5	NÃO	MRE 3	Compreensão mais aprofundada	30/30
18	0	5	NÃO	MRE 3	Compreensão mais aprofundada	30/30
19	1	5	NÃO	MRE 3	Compreensão mais aprofundada	30/30
20	4	5	NÃO	MRE 4	Compreensão mais aprofundada	30/30
21	X	5	NÃO	MRE 6	Compreensão mais aprofundada	30/30
22	6	5	SIM			30/30
23	(em branco)	6	NÃO	MRE 5	Restrição de Interpretação	30/30
24	0	6	NÃO	MRE 5	Restrição de Interpretação	30/30
25	X	6	NÃO	MRE 6	Compreensão mais aprofundada	30/30
26	1	6	SIM			30/30

Tabela 6.2: Relação de ações que simulam a resolução do exercício.

6.3 Estudo de Caso para o critério “Sucessos Anteriores com a MRE”

Foram desativadas todas as remediações referentes à meta nº 5, e cadastrada uma remediação para esta meta com os seguintes parâmetros: tratamento de resposta “qualquer erro”, tipo de erro “Diretamente Identificáveis”, subtipo de erro “Deficiência de regra, teorema ou definição” que resulta na função MRE “Compreensão mais aprofundada” e critério “Sucessos anteriores com a MRE”, o que apresenta como MREs candidatas as MREs 3, 4 e 6. Esta configuração foi escolhida para verificar o critério de sucessos anteriores e pela função MRE possuir três MREs disponíveis.

Conforme visto na tabela 6.2, as MREs 3 e 4 não obtiveram sucesso na ação seguida às suas remediações, indicando que os alunos cometeram erros após a exibição destas MREs. A MRE que apresenta sucesso na remediação foi a MRE 6, devido ao fato de os estudantes acertarem após a sua exibição. Por isso é esperado que a simulação retorne a MRE 6 para o estudante que errar no alcance da meta nº 5.

Este segundo teste foi executado 30 vezes, simulando 30 estudantes, com 7 interações cada estudante, totalizando 210 interações, que saberão resolver as demais metas e cometerão o deslize na meta nº 5. Conforme visto na tabela 6.3, a MRE esperada (MRE 6) retornou para todas as execuções de remediação desta meta.

Ação	Resposta	Meta	Resposta correta?	MRE	Função MRE	Execuções
27	1	1	SIM			30/30
28	0	2	SIM			30/30
29	6	3	SIM			30/30
30	+	4	SIM			30/30
31	X	5	NÃO	MRE 6	Compreensão Mais Aprofundada	30/30
32	6	5	SIM			30/30
33	1	6	SIM			30/30

Tabela 6.3: Relação de ações que simulam a resolução do exercício após edição de remediações.

6.4 Estudo de caso para o critério “não mostrar MRE”

A configuração escolhida para avaliar as saídas para este critério são bem parecidas com a configuração escolhida para análise do critério “Sucessos anteriores”, visto na seção 6.3. Para a análise deste critério também foram desativadas todas as remediações referentes à meta nº 5 e cadastrada uma remediação para esta meta com os seguintes parâmetros: tratamento de resposta “qualquer erro”, tipo de erro “Diretamente Identificáveis”, subtipo de erro “Deficiência de regra, teorema ou definição” que resulta na função MRE “Compreensão mais aprofundada” e critério “Não mostrar MRE”, o que apresenta como MREs candidatas as MREs 3, 4 e 6.

Este critério exige a escolha de uma MRE, por isso foi escolhida a MRE 6 para não ser exibida nesta etapa de resolução, caso o estudante cometa um erro. Com isso, espera-se que o aprendiz tenha como saída ou a MRE 3 ou a MRE 4.

Este terceiro teste procurou seguir conforme os testes anteriores, também foi executado 30 vezes, simulando 30 estudantes, com 8 interações cada estudante, totalizando 240 interações, que saberão resolver as demais metas e cometerão dois deslizos na meta nº 5. Porém, como neste caso há duas regras na Base de Regras de MREs que podem ser disparadas (uma para a MRE 3 e outra para a MRE 4), existe um conflito (ocorre quando mais de uma regra pode ser disparada) e o próprio motor de inferência do Jeops pode resolver sem a interferência do autor.

Na implementação de todas as bases de regras, optou-se não selecionar nenhuma resolução de conflito das disponíveis no motor de inferência. Com isso, esperava-se que houvesse uma certa alternância entre estas regras, mas cada teste manteve a mesma regra em todas as suas execuções. Ou seja, quando o motor de inferência disparou a regra que retorna da MRE 3, todas as execuções subsequentes retornaram a mesma regra.

Feita esta observação, foram testadas as resoluções de conflitos disponíveis no motor de inferência. E, mesmo para estas opções, cada teste manteve a mesma regra, dando

preferência para a regra que retorna a MRE 3, por ter sido a primeira inserida na base de regras. Somente o conflito LRUConflictSet (no Jeops significa “dê preferência à regra menos utilizada na base”) ativou a regra que retorna a MRE 4, mas também para todas as 30 execuções.

Conforme visto na tabela 6.4, a MREs esperadas (MRE 3 ou MRE 4) retornaram para todas as execuções de remediação desta meta, ficando de fora a MRE 6.

Ação	Resposta	Meta	Resposta correta?	MRE	Função MRE	Execuções
34	1	1	SIM			30/30
35	0	2	SIM			30/30
36	6	3	SIM			30/30
37	+	4	SIM			30/30
38	X	5	NÃO	MRE 3 ou MRE 4	Compreensão Mais Aprofundada	30/30
39	Y	5	NÃO	MRE 3 ou MRE 4	Compreensão Mais Aprofundada	30/30
40	6	5	SIM			30/30
41	1	6	SIM			30/30

Tabela 6.4: Relação de ações que simulam a resolução do exercício após edição de remediações, para estudo do critério “não mostrar MRE específica”.

Com esta situação, foi criado um resolvedor de conflito que não dê preferência para a primeira regra adicionada na base, mas faça uma escolha randômica nesses casos com mais de uma regra que possa ser acionada. Feito isso, o teste acima foi refeito e, conforme indicado na tabela 6.5, a remediação deu preferência para a regra que retorna a MRE 4, ao aparecer 20 vezes para cada deslize do estudante.

Ação	Resposta	Meta	Resposta correta?	MRE	Função MRE	Execuções
34	1	1	SIM			30/30
35	0	2	SIM			30/30
36	6	3	SIM			30/30
37	+	4	SIM			30/30
38	X	5	NÃO	MRE 3	Compreensão Mais Aprofundada	10/30
38	X	5	NÃO	MRE 4	Compreensão Mais Aprofundada	20/30
39	Y	5	NÃO	MRE 3	Compreensão Mais Aprofundada	10/30
39	Y	5	NÃO	MRE 4	Compreensão Mais Aprofundada	20/30
40	6	5	SIM			30/30
41	1	6	SIM			30/30

Tabela 6.5: Relação de ações que simulam a resolução do exercício após edição de remediações, para estudo do critério “não mostrar MRE específica” e após inclusão de escolha aleatória.

6.5 Estudo de caso para o critério “persistência no erro”

Este teste seguiu conforme os testes mencionados acima, a inclusão de uma remediação e única ativa na meta nº 5 com os mesmos parâmetros de classificação de erro e de função MRE (tratamento de resposta “qualquer erro”, tipo de erro “Diretamente Identificáveis”, subtipo de erro “Deficiência de regra, teorema ou definição” que resulta na função MRE “Compreensão mais aprofundada”).

Critério de seleção da MRE “persistência no erro”, o qual exige a especificação do número de tentativas e da MRE que deverá ser exibida nesta tentativa, que neste caso também apresenta como MREs candidatas as MREs 3, 4 e 6. Para o teste a MRE 6 deverá ser exibida após 3 tentativas do estudante.

Este terceiro teste procurou seguir conforme os testes anteriores, também foi executado 30 vezes, simulando 30 estudantes, com 9 interações cada estudante, totalizando 270 interações, que saberão resolver as demais metas e cometerão três deslizes na meta nº 5.

Conforme o esperado, a MRE 6 foi exibida na terceira tentativa para todas as execuções. A MRE 3 apareceu para as duas tentativas anteriores, conforme visto na tabela 6.6, porém poderia ocorrer a MRE 4 durante as execuções. Esta questão sobre a exibição da MRE 4 foi vista na seção 6.4.

Ação	Resposta	Meta	Resposta correta?	MRE	Função MRE	Execuções
42	1	1	SIM			30/30
43	0	2	SIM			30/30
44	6	3	SIM			30/30
45	+	4	SIM			30/30
46	X	5	NÃO	MRE 3	Compreensão Mais Aprofundada	30/30
47	Y	5	NÃO	MRE 3	Compreensão Mais Aprofundada	30/30
48	Z	5	NÃO	MRE 6	Compreensão Mais Aprofundada	30/30
49	6	5	SIM			30/30
50	1	6	SIM			30/30

Tabela 6.6: Relação de ações que simulam a resolução do exercício após edição de remediações, para o critério “persistência no erro”.

Com a implementação do resolvidor de conflito randômico de regras, a tabela 6.7 mostra que a inferência das regras procurou selecionar mais a MRE 4, porém na terceira tentativa,

continuou a exibir a MRE 6 conforme o esperado pela remediação.

Ação	Resposta	Meta	Resposta correta?	MRE	Função MRE	Execuções
42	1	1	SIM			30/30
43	0	2	SIM			30/30
44	6	3	SIM			30/30
45	+	4	SIM			30/30
46	X	5	NÃO	MRE 3	Compreensão Mais Aprofundada	12/30
46	X	5	NÃO	MRE 4	Compreensão Mais Aprofundada	18/30
47	Y	5	NÃO	MRE 3	Compreensão Mais Aprofundada	4/30
47	Y	5	NÃO	MRE 4	Compreensão Mais Aprofundada	26/30
48	Z	5	NÃO	MRE 6	Compreensão Mais Aprofundada	30/30
49	6	5	SIM			30/30
50	1	6	SIM			30/30

Tabela 6.7: Relação de ações que simulam a resolução do exercício após edição de remediações, para o critério “persistência no erro” e resolução aleatória de conflitos.

6.6 Estudo de caso para o critério “Alternância entre as MREs”

Dando sequência às mesmas configurações das seções anteriores, em uma nova e única remediação para a meta nº 5, com o critério que alterna entre as MREs pertencentes a uma Função MRE, no caso a Função Papéis Complementares que possui como MREs as MRE 3, MRE 4 e MRE 6.

O teste também resultou em 9 interações para cada execução, de um total de 30 execuções e 270 interações, ao considerar que os estudantes cometeriam o equívoco duas vezes ao tentar solucionar a meta nº 5. Na primeira execução, ao verificar a base de dados das ações anteriores, foi indicado que a última MRE utilizada para remediação foi a MRE 6.

Esta informação é utilizada para as regras deste critério de seleção, que indicam como regras acionáveis as que retornam as MREs 3 e 4. O motor de inferência resolveu o conflito ao ativar a regra que retorna a MRE 3, fazendo com que na sequência das execuções houvesse uma alternância uniforme entre as regras que retornam a MRE 3 ou MRE 4, mesmo existindo o conflito que possui a regra que retorne a MRE 6 também possa ser acionada. Isso explica-se pela mesma situação relatada na seção 6.4, o qual o motor de inferência dá prioridade à regra incluída primeiro na base.

A tabela 6.8 apresenta as ações e a quantidade de execuções de cada ação e retornos das MREs, mostrando a alternância entre as MREs 3 e 4.

Ação	Resposta	Meta	Resposta correta?	MRE	Função MRE	Execuções
51	1	1	SIM			30/30
52	0	2	SIM			30/30
53	6	3	SIM			30/30
54	+	4	SIM			30/30
55	X	5	NÃO	MRE 3	Papéis Complementares	15/30
55	X	5	NÃO	MRE 4	Papéis Complementares	15/30
56	Y	5	NÃO	MRE 3	Papéis Complementares	15/30
56	Y	5	NÃO	MRE 4	Papéis Complementares	15/30
57	Z	5	NÃO	MRE 3	Papéis Complementares	15/30
57	Z	5	NÃO	MRE 4	Papéis Complementares	15/30
58	6	5	SIM			15/30
58	6	5	SIM			15/30
59	1	6	SIM			30/30

Tabela 6.8: Relação de ações que simulam a resolução do exercício após edição de remediações para o critério “Alternância entre as MREs”.

Ao observar este cenário, foi utilizada a resolução de conflito aleatória conforme os exemplos acima, fazendo com que a MRE 6 também fosse exibida nas interações e houve uma maior uniformidade de exibição entre as três MREs, conforme visto na tabela 6.9.

Ação	Resposta	Meta	Resposta correta?	MRE	Função MRE	Execuções
51	1	1	SIM			30/30
52	0	2	SIM			30/30
53	6	3	SIM			30/30
54	+	4	SIM			30/30
55	X	5	NÃO	MRE 3	Papéis Complementares	5/30
55	X	5	NÃO	MRE 4	Papéis Complementares	15/30
55	X	5	NÃO	MRE 6	Papéis Complementares	10/30
56	Y	5	NÃO	MRE 3	Papéis Complementares	8/30
56	Y	5	NÃO	MRE 4	Papéis Complementares	7/30
56	Y	5	NÃO	MRE 6	Papéis Complementares	15/30
57	Z	5	NÃO	MRE 3	Papéis Complementares	15/30
57	Z	5	NÃO	MRE 4	Papéis Complementares	6/30
57	Z	5	NÃO	MRE 6	Papéis Complementares	9/30
58	6	5	SIM			15/30
58	6	5	SIM			15/30
59	1	6	SIM			30/30

Tabela 6.9: Relação de ações que simulam a resolução do exercício após edição de remediações para o critério “Alternância entre as MREs” com aleatoriedade na resolução de conflitos entre as regras.

6.7 Conclusões

Foi apresentado um estudo de caso com a utilização de uma interface de autoria de um exercício e a simulação de um objeto de aprendizagem que interagiu com a arquitetura de

remediação de erros e, conseqüentemente, com as regras de produção desta arquitetura que são o objetivo principal desta autoria. A interface de autoria efetua a criação do exercício ao permitir o mapeamento de um caminho de resolução com metas a serem cumpridas e remediações para eventuais erros do estudante. Desta forma, um autor sem conhecimentos de regras de produção e programação pode ser capaz de gerar e editar estas regras.

Conforme citado anteriormente, o objeto de aprendizagem seguiu o exemplo de aplicação no campo da aritmética proposto por Leite em sua tese (24), o qual já possui uma classificação de erro que efetua a relação dos tipos de erros com as funções da MRE da taxonomia proposta por Ainsworth (2), além de propor as MREs necessárias para o exemplo, que foram aproveitadas no estudo de caso.

Houve a necessidade de criar uma remediação para cada meta, com a finalidade de prever possíveis respostas do estudante não esperadas e garantir que um erro do estudante retorne sempre uma MRE. Esta remediação gera regras de produção que são acionadas quando uma resposta errada não atenda a nenhuma das mapeadas pelo autor. Para esta remediação foi utilizado o tipo de erro chamado “Solução Não Categorizável”, também útil para respostas previstas mas sem serem categorizadas. Este tipo de erro por sua vez é relacionado à função “Compreensão Mais Aprofundada”. Pode ser uma boa prática de autoria incluir este tipo de erro em futuros classificadores.

Os critérios que envolvem histórico de ações, como “sucessos anteriores com a MRE” podem não ser muito úteis no início de treinamentos, devido à falta do próprio histórico de ações do estudante (base de ações vazia) e a arquitetura não conseguir efetuar a inferência da melhor MRE.

Uma alternativa além da aleatoriedade de escolha de uma MRE seria a possibilidade de combinações entre critérios de remediações para uma mesma meta e uma mesma resposta (ou tratamento para qualquer resposta), pois se caso um critério falhar na escolha de uma

MRE, outro critério seja selecionado, ou possibilitar que um critério tenha prioridade em uma etapa inicial e após um certo treinamento, outro critério tenha preferência.

Percebeu-se a necessidade de criar uma resolução de conflitos porque o motor de inferência efetua um retorno determinístico em seu resolvidor de conflitos padrão, ao retornar sempre a primeira regra de produção inserida na base, mesmo este motor de inferência tendo como afirmativa que as regras retornariam aleatoriamente.

A complexidade de uma MRE interfere no entendimento do estudante, por isso a definição melhor deste critério é necessária, devido a várias interpretações possíveis. Por exemplo, se haverá uma alternância entre MREs de complexidades diferentes, ou exibição de MREs mais complexas para menos complexas caso o estudante cometa equívocos com certa frequência, ou se dará prioridade às MREs menos complexas nas primeiras etapas e gradualmente a apresentação de MREs mais complexas conforme o avanço nas atividades.

As regras de produção acionadas por cada interação com o objeto de aprendizagem vistas na tabela 6.2 podem ser vistas no Apêndice A, em uma execução completa.

CAPÍTULO 7

CONSIDERAÇÕES FINAIS

O presente trabalho apresentou uma proposta de autoria de regras de produção para uma arquitetura de remediação de erros matemáticos através de múltiplas representações externas, chamado ARPTEM. Porém, a autoria de arquivos de regras de produção não impede o procedimento de rotina de programadores, podendo estes arquivos serem editados em editores de texto, caso o usuário já tenha conhecimentos de regras de produção e conhecimentos de programação. Levando em conta esta situação, a proposta consistiu em apresentar uma forma de autoria que possibilite usuários, sem conhecimentos de regras de produção e programação, manipular e gerar estas regras de produção para remediação de erros dentro da arquitetura.

Autoria feita primeiramente por uma interface de simulação de um exercício e a exemplificação de sua resolução, que gerou um caminho de resolução com os passos que indicam o estado dos valores dos componentes da interface. Com isso, o autor poderá efetuar a inclusão de erros previstos para cada etapa do caminho de resolução e, conseqüentemente, inserir remediações, através de uma segunda interface. Como a arquitetura de remediação de erros envolve mais de uma base de regras de produção, esta segunda interface, orientada a menus, permite ao autor a criação da remediação conforme o modelo proposto por Leite (24), além de gerar e gerenciar todas as regras de produção envolvidas com esta remediação, ao utilizar um classificador de erros pré-definido (já cadastrado) e MREs já com suas funções da taxonomia de Ainsworth (1). No processo de criação e edição da remediação há um Controlador de Definição com a finalidade de garantir que as regras, se acionadas, retornem o resultado esperado, no caso uma MRE.

Além das regras geradas, os dados da remediação são salvos em uma base própria para futura edição ou remoção das regras, e todas elas utilizam um identificador desta

remediação. Foi necessário incluir na autoria o armazenamento destas informações da remediação via banco de dados, para efetuar o correto controle das informações das regras de produção em sua edição, exclusão ou mesmo no momento da criação.

Analisando a ferramenta final percebem-se alguns métodos observados por Murray em seu trabalho para a autoria de STIs (37): visualização do caminho de resolução através de um diagrama de estados permite a visualização do conhecimento; interface de remediação orientada a menus permite o gerenciamento da linha de trabalho do autor; reutilização do conhecimento ao aproveitar MREs, classificadores de outros especialistas, regras de produção e exercícios de outros usuários; informações consistentes passadas pelo autor, ao utilizar o Controlador de Definição para garantir uma remediação completa. Estes métodos podem indicar ganhos em usabilidade e sugerir que a ferramenta permite facilidade na manipulação de regras diminuição dos esforços (em tempo, custo e/ou outros recursos) se comparada a autoria por código direto.

Com a simulação de ações feitas no objeto de aprendizagem, é sugerido que a autoria das regras de produção obteve resultados satisfatórios, visto que as MREs retornadas foram as esperadas em todas as execuções. Porém, apesar de esta proposta proporcionar a utilização de autores sem domínio de regras de produção com redução da carga cognitiva, são esperados dos autores, a priori, conhecimentos sobre as MREs, suas funções da taxonomia e conhecimentos de classificação de erros, para uma autoria eficaz com relação à remediações que atendam satisfatoriamente às necessidades dos estudantes.

Devido a esta exigência prévia de conhecimentos por parte dos professores, já que o trabalho focou principalmente na autoria de regras de produção, alguns leitores podem enxergar o produto deste trabalho como uma ferramenta de projeto de OAs, ficando o conceito de autoria de fato mais para as bases de regras de produção. Por isso é necessária uma avaliação da ARPREM por parte de especialistas e professores para melhor indicação desses conceitos.

A taxonomia de MREs de Ainsworth foi utilizada neste trabalho por dar continuidade ao trabalho de Leite. Porém, outras taxonomias podem ser utilizadas, como a taxonomia de Bloom (17) apud (9) que envolvam os domínios cognitivo (utilizado pelas MREs), afetivo e psicomotor. Estudos para remediação de erros com outras taxonomias podem ser interessantes.

7.1 Trabalhos Futuros

Tanto a arquitetura de autoria das regras de produção quanto a estrutura de simulação do exercício podem possibilitar alguns aperfeiçoamentos, além de mais investigações, análises e a contribuição destas. Autoria de exercícios de outros domínios matemáticos, além de inclusão de mais de um caminho de resolução que já é previsto neste modelo.

Modificações nas estruturas das regras de produção ou criação de novos tipos de estruturas serão necessárias ao serem modelados e incorporados novos critérios de seleção da melhor MRE.

Outras formas de autoria de exercícios, que não sigam a abordagem baseada em exemplos, de rastreamento do estudante ou que tracem todo o caminho de resolução podem ser pensadas, como a modelagem por restrições (34). Caso se pense na implementação de outras formas de autoria, estas podem ser avaliadas em paralelo para a análise da abordagem mais eficaz com relação aos esforços no processo de autoria ou com relação às saídas retornadas.

Ou simplesmente há a possibilidade de uma autoria de regras sem a interface de simulação de um exercício como apoio. Porém, também poderá afetar a criação de novos arranjos estruturais das regras de produção e demandará em novas composições de interface.

A arquitetura levou em conta o envolvimento de um campo para cada caminho de resolução, o que pode ser cansativo para o autor em OAs mais complexos. Por isso pode-se pensar na possibilidade de envolver mais de um campo da Interface do OA em um

mesmo caminho de resolução, conforme as necessidades do autor.

Além das remediações que são incluídas em cada passo de resolução, a possibilidade de incluir remediações gerais, que independem da etapa que o aprendiz se encontra para que suas regras sejam acionadas. Com isso pode facilitar a autoria no processo de inclusão das regras e evitar repetição de remediações.

A inclusão de conhecimento incorporado (37) para auxiliar os autores na manipulação das informações, como disponibilidade de apoio ao cadastro de classificadores de erros e compreensão sobre as funções das MREs, pode contribuir para uma autoria satisfatória.

O histórico de ações dos estudantes podem proporcionar uma vasta análise desses dados, por exemplo:

- MREs que mais contribuíram para a remediação dos estudantes;
- quais os caminhos de resolução de um exercício apresenta mais dificuldade ou facilidade por parte dos estudantes;
- os impactos das formas de MRE (gráfico, texto, ...) e das funções das MREs em um exercício ou em um grupo de estudantes;
- análise da complexidade das MREs;
- base das ações pode ser utilizada para efetuar pesquisas que envolvam aprendizagem de máquina, para exercícios mais complexos;
- base das ações pode contribuir também para futuras pesquisas que envolvam autoria inteligente deste modelo, ao indicar para o autor no momento da criação de remediação qual MRE candidata mais promissora, ou qual a melhor função MRE para o tipo de erro em questão;
- análise dos melhores critérios de seleção de uma MRE;
- traçar perfil do estudante para domínios semelhantes.

Também é importante uma avaliação qualitativa desta ferramenta, bem como das interfaces desenvolvidas, por parte de especialistas dos domínios envolvidos.

Além dos itens citados anteriormente, a arquitetura pode ser adaptada para as subfunções das funções das MREs.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Shaaron Ainsworth. The functions of multiple representations. 33:131–152, 1999.
- [2] Shaaron Ainsworth. Deft: A conceptual framework for learning with multiple representations learning and instruction. *Learning and Instruction*, 16(3):183–198, 2006.
- [3] Shaaron Ainsworth. The educational value of multiple-representations when learning complex scientific concepts. *Theory and practice in science education*, páginas 191–208, 2008.
- [4] Vincent Aleven. Rule-Based Cognitive Modeling for Intelligent Tutoring Systems. Roger Nkambou, Jacqueline Bourdeau, e Riichirio Mizoguchi, editors, *Studies in Computational Intelligence*, 308, páginas 33–62. Springer-Verlag Berlin Heidelberg, 2010.
- [5] Vincent Aleven, Bruce M. McLaren, Jonathan Sewall, e Kenneth R. Koedinger. The cognitive tutor authoring tools (ctat): Preliminary evaluation of efficiency gains. páginas 61–70, 2006.
- [6] J. Anderson. *The architecture of cognition*. Harvard University Press, 1983.
- [7] Nick Bassiliades, Efstratios Kontopoulos, Grigoris Antoniou, e Ioannis Vlahavas. A graphical rule authoring tool for defeasible reasoning in the semantic web. *Springer*, páginas 404–414, 2005.
- [8] Stephen B. Blessing. A Programming By Demonstration Authoring Tool For Model-Tracing Tutors. *International Journal of Artificial Intelligence in Education*, 1997.
- [9] B. S. Bloom et al. *Taxonomy of educational objectives*, volume 1. David McKay, 1956.
- [10] Sanket D. Choksey. Developing an Affordable Authoring Tool for Intelligent Tutoring Systems. Dissertação de Mestrado, Worchester Polytechnic Institute, August de 2004.

- [11] P. R. Cohen. The Role Of Natural Language in a Multimodal Interface. 504. SRI International, 1991.
- [12] Richard Cox e Paul Brna. Supporting the use of external representations in problem solving: The need for flexible learning environments. *Journal of Artificial Intelligence in Education*, 6:239–302, 1995.
- [13] A. Cypher e D. C. Halbert. *Watch what I do: Programming by Demonstration*. The MIT Press, 1993.
- [14] Carlos Santos da Figueira Filho e Geber Lisboa Ramalho. Jeops – the java embedded object production system. páginas 52–61, 2000.
- [15] Fábio Duarte de Oliveira. Suporte ao aprendizado apoiado por múltiplas representações externas através da análise e remediação de erros. Dissertação de Mestrado, Universidade Federal do Paraná, 2011.
- [16] Alexandre Ibrahim Direne. *Metodology and Tools for Designing Concept Tutoring Systems*. Tese de Doutorado, July de 1993.
- [17] A. P. C. M. Ferraz e R. V. Belhot. Taxonomia de Bloom: revisão teórica e apresentação das adequações do instrumento para definição de objetivos instrucionais. *Gest. Prod. São Carlos*, 17(2):421–431, 2010.
- [18] Laura Sánchez García. *LINUX: Um ambiente integrado de interface para sistemas de informação baseados em conhecimento*. Tese de Doutorado, 1995.
- [19] S. R. Goldman. Learning in complex domains: When and why do multiple representations help? *Learning and Instruction*, 13(2):239–244, 2003.
- [20] Matthew Paul Jarvis. Applying Machine Learning Techniques to Rule Generation in Intelligent Tutoring Systems. Dissertação de Mestrado, maio de 2004.
- [21] Kenneth R. Koedinger, Vincent Aleven, e Neil Heffernan. Toward a rapid development environment for cognitive tutors. Relatório técnico, 2003.

- [22] Eleandro Maschio Krynski. Uma abordagem metacognitiva através de múltiplas representações externas para o ensino de programação de computadores. Dissertação de Mestrado, 2007.
- [23] Rohit Kumar, Matthew E. Roy, R. Bruce Roberts, e John I. Makhoul. Towards automatically building tutor models using multiple behavior demonstrations. *ITS*, páginas 535–544, 2014.
- [24] Maici Duarte Leite. *Arquitetura para Remediação de Erros Baseada em Múltiplas Representações Externas*. Tese de Doutorado, Universidade Federal do Paraná, 2013.
- [25] Maici Duarte Leite, Andrey Ricardo Pimentel, e Mônica Hoeldtke Pietrichinski. Remediação de erros baseada em múltiplas representações externas e classificação de erros aplicada a objetos de aprendizagem inteligentes. *Anais do 23º Simpósio Brasileiro de Informática na Educação*, 2012.
- [26] H. Lieberman. *Your Wish is My Command: Programming by Example*. Morgan Kaufmann Publishers Inc., 2001.
- [27] Christopher J. MacLellan, Kenneth R. Koedinger, e Noboru Matsuda. Authoring Tutors with SimStudent: An Evaluation of Efficiency and Model Quality. *ITS*, páginas 551–560, 2014.
- [28] D. Marczal e A. I. Direne. Um arcabouço que enfatiza a retroação a contextos de erro na solução de problemas. *Revista Brasileira de Informática na Educação*, 19(63-73):19–27, 2011.
- [29] D. Marczal e A. I. Direne. Farma: Uma ferramenta de autoria para objetos de aprendizagem de conceitos matemáticos. *Anais do 23º Simpósio Brasileiro de Informática na Educação*, páginas 249–255, 2012.
- [30] Diego Marczal. Um arcabouço que enfatiza a retroação a contextos de erro durante o acesso a conteúdos educacionais. Dissertação de Mestrado, 2010.

- [31] Diego Marczal. *FARMA: Uma ferramenta de autoria para objetos de aprendizagem de conceitos matemáticos*. Tese de Doutorado, 2014.
- [32] Noboru Matsuda, William W. Cohen, e Kenneth R. Koedinger. An intelligent authoring system with programming by demonstration. *Proceedings of the Japan National Conference on Information and Systems in Education*, 2005.
- [33] Noboru Matsuda, William W. Cohen, e Kenneth R. Koedinger. Teaching the teacher: Tutoring simstudent leads to more effective cognitive tutor authoring. *International Artificial Intelligence in Education Society*, páginas 34, may de 2014.
- [34] Antonija Mitrovic. Modelling domains and students with constraint based-modeling. Roger Nkambou, Jacqueline Bourdeau, e Riichirio Mizoguchi, editors, *Studies in Computational Intelligence*, 308, páginas 63–80. Springer-Verlag Berlin Heidelberg, 2010.
- [35] L. A. Munárriz. *Fundamentos da Inteligência Artificial*. 1994.
- [36] Tom Murray. Authoring knowledge based tutors: Tools for content, instructional strategy, student model, and interface design. *Journal of the Learning Sciences*, 7(1):5–64, 1998.
- [37] Tom Murray. Authoring intelligent tutoring systems: An analysis of the state of the art. *International Journal of Artificial Intelligence in Education*, 10:98–129, 1999.
- [38] Tom Murray et al. An overview of intelligent tutoring system authoring tools: Updated analysis of state of art. Tom Murray, Shaaron Ainsworth, Stephen Blessing, et al., editors, *Authoring Tools for Advanced Technology Learning Environments*, capítulo 17, páginas 493–546. Kluwer Academic Publishers, 2003.
- [39] Hyacinth S. Nwana. Intelligent tutoring systems: an overview. *Artificial Intelligence Review*, 4:251–277, 1990.
- [40] Jennifer K. Olsen, Vincent Aleven, Jonathan Sewall, Daniel M. Belenky, e Nikol

- Rummel. Authoring Tools for Collaborative Intelligent Tutoring System Environments. *ITS*, páginas 523–528, 2014.
- [41] S. E. Palmer. Fundamental aspects of cognitive representation. *Cognition and categorization*, 1977.
- [42] Luc Paquette, Jean-François Lebeau, Gabriel Beaulieu, e André Mayers. Automating next-step hints generation using astus. *ITS*, páginas 201–211, 2012.
- [43] Luc Paquette, Jean-François Lebeau, e André Mayers. Authoring Problem-Solving Tutors: A Comparison between ASTUS and CTAT. Roger Nkambou, Jacqueline Bourdeau, e Riichirio Mizoguchi, editors, *Studies in Computational Intelligence*, 308, páginas 377–405. Springer-Verlag Berlin Heidelberg, 2010.
- [44] Andrey Ricardo Pimentel. Medidas cognitivas para o ensino de conceitos visuais com sistemas tutores inteligentes. Dissertação de Mestrado, outubro de 1997.
- [45] Leena Razzaq, Mingyu Feng, Kenneth R. Koedinger, Neil T. Heffernan, Brian Junker, Goss Nuzzo-Jones, Michael A. Macasek, Rasmussen Kai P., Terrence E. Turner, e Jason A. Walonoski. A web-based authoring tool for intelligent tutors: Blending assessment and instructional assistance. *Studies in Computational Intelligence (SCI)*, 44:23–49, 2007.
- [46] Leena Razzaq e Neil T. Heffernan. Open-Content Authoring Tools. Roger Nkambou, Jacqueline Bourdeau, e Riichirio Mizoguchi, editors, *Studies in Computational Intelligence*, 308, páginas 407–420. Springer-Verlag Berlin Heidelberg, 2010.
- [47] Steven Ritter, John R. Anderson, Kenneth R. Koedinger, e Albert Corbett. Cognitive tutor: Applied research in mathematics education. *Psychonomic Bulletin & Review*, páginas 249–255, 2007.
- [48] A. N. Taatgen. Learning rules and productions. *Encyclopedia of Cognitive Science*, (2):822–830, 2003.

- [49] Etienne Wenger. *Artificial intelligence and tutoring systems: computational and cognitive approaches to the communication of knowledge*. Morgan Kaufmann Publishers Inc., 1987.
- [50] M. Zancanaro, A. Cappelletti, C. Signorini, e C. Strapparava. Authoring the "intelligence" of an educational game. *AAAI Technical Report*, páginas 82–86, 2001.
- [51] Jiajie Zhang. The nature of external representations in problem solving. *Cognitive Science*, 21(2):179–217, 1997.

APÊNDICE A

LISTA DAS REGRAS ACIONADAS NA EXECUÇÃO DO ESTUDO DE CASO

Listagem das regras acionadas na execução do OA de exemplo para estudo de caso, visto no capítulo 6. Cada interação é indicada por uma sequência de traços ("----") e o número que identifica a execução. Em seguida, as regras acionadas até a identificação da próxima interação.

----- interação:1

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_1
_COMPONENTE_txt1_REMEDIACAO_1_

Condições:

Resposta = " "

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_1_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_1

_COMPONENTE_txt1_REMEDIACAO_1_

Condições:

Resposta = " "

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Classifique o tipo de erro como "Interpretação Equivocada"

Regra acionada para classificar a Função da MRE:

funcaoMRE_1_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_1

_COMPONENTE_txt1_REMEDIACAO_1_

Condições:

Tipo de Erro = "Interpretação Equivocada"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Classifique a Função MRE como "Papéis Complementares"

Regra acionada para selecionar a MRE:

regraMRE_1_REMEDIACAO_1_criterion_1_

Condições:

MRE = "MRE 1"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 1"

----- interação:2

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_1

_COMPONENTE_txt1_REMEDIACAO_4_

Condições:

Resposta = "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_2_suberrortype_1_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_1

_COMPONENTE_txt1_REMEDIACAO_4_

Condições:

Resposta = "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Classifique o tipo de erro como "Diretamente Identificáveis"

e classifique o subtipo do erro como "Deficiência em relação ao domínio ou uso inadequado de dados"

Regra acionada para classificar a Função da MRE:

funcaoMRE_2_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_1

_COMPONENTE_txt1_REMEDIACAO_4_

Condições:

Tipo de Erro = "Diretamente Identificáveis"

Subtipo de Erro = "Deficiência em relação ao domínio ou uso inadequado de dados"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Classifique a Função MRE como "Restrição de Interpretação"

Regra acionada para selecionar a MRE:

regraMRE_2_REMEDIACAO_4_criterion_1_

Condições:

MRE = "MRE 2"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 2"

----- interação:3

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_1

_COMPONENTE_txt1_REMEDIACAO_5_

Condições:

Resposta = "6"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_2_suberrortype_1_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_1
_COMPONENTE_txt1_REMEDIACAO_5_

Condições:

Resposta = "6"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Classifique o tipo de erro como "Diretamente Identificáveis"

e classifique o subtipo do erro como "Deficiência em relação ao domínio ou uso inadequado de dados"

Regra acionada para classificar a Função da MRE:

funcaoMRE_2_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_1
_COMPONENTE_txt1_REMEDIACAO_5_

Condições:

Tipo de Erro = "Diretamente Identificáveis"

Subtipo de Erro = "Deficiência em relação ao domínio ou uso inadequado de dados"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Classifique a Função MRE como "Restrição de Interpretação"

Regra acionada para selecionar a MRE:

regraMRE_2_REMEDIACAO_4_criterion_1_

Condições:

MRE = "MRE 2"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 2"

----- interação:4

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_1

_COMPONENTE_txt1_REMEDIACAO_15_

Condições:

Resposta diferente de "1"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_4_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_1

_COMPONENTE_txt1_REMEDIACAO_15_

Condições:

Resposta diferente de "1"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Classifique o tipo de erro como "Solução não categorizável"

Regra acionada para classificar a Função da MRE:

funcaoMRE_3_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_1

_COMPONENTE_txt1_REMEDIACAO_15_

Condições:

Tipo de Erro = "Solução não categorizável"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Classifique a Função MRE como "Compreensão mais aprofundada"

Regra acionada para selecionar a MRE:

regraMRE_6_REMEDIACAO_15_criterion_1_

Condições:

MRE = "MRE 6"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 6"

----- interação:5

Regra acionada para resposta correta: respostaCorreta_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_1

_COMPONENTE_txt1_

Condições:

Resposta = "1"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 1

Componente = "txt1"

Ações:

Marque a Meta 1 como satisfeita

----- interação:6

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_2
_COMPONENTE_txt2_REMEDIACAO_2_

Condições:

Resposta = " "

Exercício = 1

Questão = 1

Caminho = 1

Meta = 2

Componente = "txt2"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_1_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_2
_COMPONENTE_txt2_REMEDIACAO_2_

Condições:

Resposta = " "

Exercício = 1

Questão = 1

Caminho = 1

Meta = 2

Componente = "txt2"

Ações:

Classifique o tipo de erro como "Interpretação Equivocada"

Regra acionada para classificar a Função da MRE:

funcaoMRE_1_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_2
_COMPONENTE_txt2_REMEDIACAO_2_

Condições:

Tipo de Erro = "Interpretação Equivocada"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 2

Componente = "txt2"

Ações:

Classifique a Função MRE como "Papéis Complementares"

Regra acionada para selecionar a MRE:

regraMRE_1_REMEDIACAO_1_criterion_1_

Condições:

MRE = "MRE 1"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 1"

----- interação:7

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_2

_COMPONENTE_txt2_REMEDIACAO_6_

Condições:

Resposta = "6"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 2

Componente = "txt2"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_2_suberrortype_1_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_2

_COMPONENTE_txt2_REMEDIACAO_6_

Condições:

Resposta = "6"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 2

Componente = "txt2"

Ações:

Classifique o tipo de erro como "Diretamente Identificáveis"

e classifique o subtipo do erro como "Deficiência em relação ao domínio ou uso inadequado de dados"

Regra acionada para classificar a Função da MRE:

funcaoMRE_2_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_2
_COMPONENTE_txt2_REMEDIACAO_6_

Condições:

Tipo de Erro = "Diretamente Identificáveis"

Subtipo de Erro = "Deficiência em relação ao domínio ou uso inadequado de dados"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 2

Componente = "txt2"

Ações:

Classifique a Função MRE como "Restrição de Interpretação"

Regra acionada para selecionar a MRE:

regraMRE_2_REMEDIACAO_4_criterion_1_

Condições:

MRE = "MRE 2"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 2"

----- interação:8

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_2
_COMPONENTE_txt2_REMEDIACAO_16_

Condições:

Resposta diferente de "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 2

Componente = "txt2"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_4_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_2
_COMPONENTE_txt2_REMEDIACAO_16_

Condições:

Resposta diferente de "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 2

Componente = "txt2"

Ações:

Classifique o tipo de erro como "Solução não categorizável"

Regra acionada para classificar a Função da MRE:

funcaoMRE_3_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_2
_COMPONENTE_txt2_REMEDIACAO_16_

Condições:

Tipo de Erro = "Solução não categorizável"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 2

Componente = "txt2"

Ações:

Classifique a Função MRE como "Compreensão mais aprofundada"

Regra acionada para selecionar a MRE:

regraMRE_6_REMEDIACAO_15_criterion_1_

Condições:

MRE = "MRE 6"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 6"

----- interação:9

Regra acionada para resposta correta: respostaCorreta_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_2

_COMPONENTE_txt2_

Condições:

Resposta = "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 2

Componente = "txt2"

Ações:

Marque a Meta 2 como satisfeita

----- interação:10

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_3

_COMPONENTE_txt5_REMEDIACAO_3_

Condições:

Resposta = "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

*tipoErro_2_suberrortype_1_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_3
_COMPONENTE_txt5_REMEDIACAO_3_*

Condições:

Resposta = "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Classifique o tipo de erro como "Diretamente Identificáveis"

e classifique o subtipo do erro como "Deficiência em relação ao domínio ou uso inadequado de dados"

Regra acionada para classificar a Função da MRE:

*funcaoMRE_2_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_3
_COMPONENTE_txt5_REMEDIACAO_3_*

Condições:

Tipo de Erro = "Diretamente Identificáveis"

Subtipo de Erro = "Deficiência em relação ao domínio ou uso inadequado de dados"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Classifique a Função MRE como "Restrição de Interpretação"

Regra acionada para selecionar a MRE:

regraMRE_2_REMEDIACAO_4_criterion_1_

Condições:

MRE = "MRE 2"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 2"

----- interação:11

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_3

_COMPONENTE_txt5_REMEDIACAO_7_

Condições:

Resposta = "1"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_2_suberrortype_1_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_3
_COMPONENTE_txt5_REMEDIACAO_7_

Condições:

Resposta = "1"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Classifique o tipo de erro como "Diretamente Identificáveis"

e classifique o subtipo do erro como "Deficiência em relação ao domínio ou uso inadequado de dados"

Regra acionada para classificar a Função da MRE:

funcaoMRE_2_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_3
_COMPONENTE_txt5_REMEDIACAO_7_

Condições:

Tipo de Erro = "Diretamente Identificáveis"

Subtipo de Erro = "Deficiência em relação ao domínio ou uso inadequado de dados"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Classifique a Função MRE como "Restrição de Interpretação"

Regra acionada para selecionar a MRE:

regraMRE_2_REMEDIACAO_4_criterion_1_

Condições:

MRE = "MRE 2"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 2"

----- interação:12

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_3

_COMPONENTE_txt5_REMEDIACAO_17_

Condições:

Resposta diferente de "6"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_4_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_3

_COMPONENTE_txt5_REMEDIACAO_17_

Condições:

Resposta diferente de "6"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Classifique o tipo de erro como "Solução não categorizável"

Regra acionada para classificar a Função da MRE:

funcaoMRE_3_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_3

_COMPONENTE_txt5_REMEDIACAO_17_

Condições:

Tipo de Erro = "Solução não categorizável"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Classifique a Função MRE como "Compreensão mais aprofundada"

Regra acionada para selecionar a MRE:

regraMRE_6_REMEDIACAO_15_criterion_1_

Condições:

MRE = "MRE 6"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 6"

----- interação:13

Regra acionada para resposta correta: respostaCorreta_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_3

_COMPONENTE_txt5_

Condições:

Resposta = "6"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 3

Componente = "txt5"

Ações:

Marque a Meta 3 como satisfeita

----- interação:14

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_4

_COMPONENTE_txt3_REMEDIACAO_8_

Condições:

Resposta = "-"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 4

Componente = "txt3"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_2_suberrortype_3_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_4

_COMPONENTE_txt3_REMEDIACAO_8_

Condições:

Resposta = "-"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 4

Componente = "txt3"

Ações:

Classifique o tipo de erro como "Diretamente Identificáveis"

e classifique o subtipo do erro como "Deficiência na escolha do operador correto"

Regra acionada para classificar a Função da MRE:

funcaoMRE_3_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_4

_COMPONENTE_txt3_REMEDIACAO_8_

Condições:

Tipo de Erro = "Diretamente Identificáveis"

Subtipo de Erro = "Deficiência na escolha do operador correto"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 4

Componente = "txt3"

Ações:

Classifique a Função MRE como "Compreensão mais aprofundada"

Regra acionada para selecionar a MRE:

regraMRE_4_REMEDIACAO_8_criterion_1_

Condições:

MRE = "MRE 4"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 4"

----- interação:15

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_4

_COMPONENTE_txt3_REMEDIACAO_18_

Condições:

Resposta diferente de "+"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 4

Componente = "txt3"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_4_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_4

_COMPONENTE_txt3_REMEDIACAO_18_

Condições:

Resposta diferente de "+"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 4

Componente = "txt3"

Ações:

Classifique o tipo de erro como "Solução não categorizável"

Regra acionada para classificar a Função da MRE:

funcaoMRE_3_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_4

_COMPONENTE_txt3_REMEDIACAO_18_

Condições:

Tipo de Erro = "Solução não categorizável"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 4

Componente = "txt3"

Ações:

Classifique a Função MRE como "Compreensão mais aprofundada"

Regra acionada para selecionar a MRE:

regraMRE_6_REMEDIACAO_15_criterion_1_

Condições:

MRE = "MRE 6"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 6"

----- *interação:16*

Regra acionada para resposta correta: respostaCorreta_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_4

_COMPONENTE_txt3_

Condições:

Resposta = "+"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 4

Componente = "txt3"

Ações:

Marque a Meta 4 como satisfeita

----- *interação:17*

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_5

_COMPONENTE_txt8_REMEDIACAO_9_

Condições:

Resposta = "4"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_2_suberrortype_3_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_5

_COMPONENTE_txt8_REMEDIACAO_9_

Condições:

Resposta = "4"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Classifique o tipo de erro como "Diretamente Identificáveis"

e classifique o subtipo do erro como "Deficiência na escolha do operador correto"

Regra acionada para classificar a Função da MRE:

funcaoMRE_3_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_5
_COMPONENTE_txt8_REMEDIACAO_9_

Condições:

Tipo de Erro = "Diretamente Identificáveis"

Subtipo de Erro = "Deficiência na escolha do operador correto"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Classifique a Função MRE como "Compreensão mais aprofundada"

Regra acionada para selecionar a MRE:

regraMRE_4_REMEDIACAO_8_criterion_1_

Condições:

MRE = "MRE 4"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 4"

----- interação:18

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_5
_COMPONENTE_txt8_REMEDIACAO_10_

Condições:

Resposta = "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_2_suberrortype_2_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_5
_COMPONENTE_txt8_REMEDIACAO_10_

Condições:

Resposta = "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Classifique o tipo de erro como "Diretamente Identificáveis"

e classifique o subtipo do erro como "Deficiência de regra, teorema ou definição"

Regra acionada para classificar a Função da MRE:

funcaoMRE_3_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_5
_COMPONENTE_txt8_REMEDIACAO_10_

Condições:

Tipo de Erro = "Diretamente Identificáveis"

Subtipo de Erro = "Deficiência de regra, teorema ou definição"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Classifique a Função MRE como "Compreensão mais aprofundada"

Regra acionada para selecionar a MRE:

regraMRE_3_REMEDIACAO_10_criterion_1_

Condições:

MRE = "MRE 3"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 3"

----- interação:19

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_5
_COMPONENTE_txt8_REMEDIACAO_11_

Condições:

Resposta = "1"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_2_suberrortype_2_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_5
_COMPONENTE_txt8_REMEDIACAO_11_

Condições:

Resposta = "1"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Classifique o tipo de erro como "Diretamente Identificáveis"

e classifique o subtipo do erro como "Deficiência de regra, teorema ou definição"

Regra acionada para classificar a Função da MRE:

funcaoMRE_3_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_5
_COMPONENTE_txt8_REMEDIACAO_11_

Condições:

Tipo de Erro = "Diretamente Identificáveis"

Subtipo de Erro = "Deficiência de regra, teorema ou definição"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Classifique a Função MRE como "Compreensão mais aprofundada"

Regra acionada para selecionar a MRE:

regraMRE_3_REMEDIACAO_10_criterion_1_

Condições:

MRE = "MRE 3"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 3"

----- interação:20

Regra acionada para

resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_5

_COMPONENTE_txt8_REMEDIACAO_12_

Condições:

Resposta = " "

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_2_suberrortype_2_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_5

_COMPONENTE_txt8_REMEDIACAO_12_

Condições:

Resposta = " "

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Classifique o tipo de erro como "Diretamente Identificáveis"

e classifique o subtipo do erro como "Deficiência de regra, teorema ou definição"

Regra acionada para classificar a Função da MRE:

funcaoMRE_3_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_5

_COMPONENTE_txt8_REMEDIACAO_12_

Condições:

Tipo de Erro = "Diretamente Identificáveis"

Subtipo de Erro = "Deficiência de regra, teorema ou definição"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Classifique a Função MRE como "Compreensão mais aprofundada"

Regra acionada para selecionar a MRE:

regraMRE_3_REMEDIACAO_10_criterion_1_

Condições:

MRE = "MRE 3"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 3"

----- interação:21

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_5
_COMPONENTE_txt8_REMEDIACAO_19_

Condições:

Resposta diferente de "6"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_4_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_5
_COMPONENTE_txt8_REMEDIACAO_19_

Condições:

Resposta diferente de "6"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Classifique o tipo de erro como "Solução não categorizável"

Regra acionada para classificar a Função da MRE:

funcaoMRE_3_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_5

_COMPONENTE_txt8_REMEDIACAO_19_

Condições:

Tipo de Erro = "Solução não categorizável"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Classifique a Função MRE como "Compreensão mais aprofundada"

Regra acionada para selecionar a MRE:

regraMRE_6_REMEDIACAO_15_criterion_1_

Condições:

MRE = "MRE 6"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 6"

----- interação:22

Regra acionada para resposta correta: respostaCorreta_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_5
_COMPONENTE_txt8_

Condições:

Resposta = "6"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 5

Componente = "txt8"

Ações:

Marque a Meta 5 como satisfeita

----- interação:23

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1
_CAMINHO_1_META_6
_COMPONENTE_txt7_REMEDIACAO_13_

Condições:

Resposta = " "

Exercício = 1

Questão = 1

Caminho = 1

Meta = 6

Componente = "txt7"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_3_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_6

_COMPONENTE_txt7_REMEDIACAO_13_

Condições:

Resposta = " "

Exercício = 1

Questão = 1

Caminho = 1

Meta = 6

Componente = "txt7"

Ações:

Classifique o tipo de erro como "Indiretamente identificáveis"

Regra acionada para classificar a Função da MRE:

funcaoMRE_2_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_6

_COMPONENTE_txt7_REMEDIACAO_13_

Condições:

Tipo de Erro = "Indiretamente identificáveis"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 6

Componente = "txt7"

Ações:

Classifique a Função MRE como "Restrição de Interpretação"

Regra acionada para selecionar a MRE:

regraMRE_5_REMEDIACAO_13_criterion_1_

Condições:

MRE = "MRE 5"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 5"

----- interação:24

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_6

_COMPONENTE_txt7_REMEDIACAO_14_

Condições:

Resposta = "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 6

Componente = "txt7"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_3_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_6

_COMPONENTE_txt7_REMEDIACAO_14_

Condições:

Resposta = "0"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 6

Componente = "txt7"

Ações:

Classifique o tipo de erro como "Indiretamente identificáveis"

Regra acionada para classificar a Função da MRE:

funcaoMRE_2_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_6

_COMPONENTE_txt7_REMEDIACAO_14_

Condições:

Tipo de Erro = "Indiretamente identificáveis"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 6

Componente = "txt7"

Ações:

Classifique a Função MRE como "Restrição de Interpretação"

Regra acionada para selecionar a MRE:

regraMRE_5_REMEDIACAO_13_criterion_1_

Condições:

MRE = "MRE 5"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 5"

----- interação:25

Regra acionada para resposta errada:

respostaErrada_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_6

_COMPONENTE_txt7_REMEDIACAO_20_

Condições:

Resposta diferente de "1"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 6

Componente = "txt7"

Ações:

Marque a resposta como errada

Regra acionada para classificar o erro:

tipoErro_4_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_6

_COMPONENTE_txt7_REMEDIACAO_20_

Condições:

Resposta diferente de "1"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 6

Componente = "txt7"

Ações:

Classifique o tipo de erro como "Solução não categorizável"

Regra acionada para classificar a Função da MRE:

funcaoMRE_3_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_6

_COMPONENTE_txt7_REMEDIACAO_20_

Condições:

Tipo de Erro = "Solução não categorizável"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 6

Componente = "txt7"

Ações:

Classifique a Função MRE como "Compreensão mais aprofundada"

Regra acionada para selecionar a MRE:

regraMRE_6_REMEDIACAO_15_criterion_1_

Condições:

MRE = "MRE 6"

Critério = "MRE específica para o erro"

Ações:

Exiba a MRE "MRE 6"

----- interação:26

Regra acionada para resposta correta: respostaCorreta_EXERCICIO_1_QUESTAO_1

_CAMINHO_1_META_6

_COMPONENTE_txt7_

Condições:

Resposta = "1"

Exercício = 1

Questão = 1

Caminho = 1

Meta = 6

Componente = "txt7"

Ações:

Marque a Meta 6 como satisfeita

LEANDRO RODRIGUES FERREIRA

**ARPREM: AUTORIA DE REGRAS DE PRODUÇÃO PARA
REMEDIÇÃO DE ERROS COM MÚLTIPLAS
REPRESENTAÇÕES EXTERNAS**

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre. Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, Universidade Federal do Paraná.

Orientador: Prof. Dr. Andrey Ricardo Pimentel

CURITIBA

2015